

305-CD-100-005

## **EOSDIS Core System Project**

# **Release 4 Segment/Design Specification for the ECS Project**

February 1999

Final

Raytheon Systems Company  
Upper Marlboro, Maryland

**Release 4  
Segment/Design Specification  
for the ECS Project**

**February 1999**

**Final**

Prepared Under Contract NAS-60000  
CDRL Item #046

**RESPONSIBLE ENGINEER**

<u>Alton Davis, Jr. /s/</u>	<u>2/11/99</u>
Alton Davis	Date
EOSDIS Core System Project	

**SUBMITTED BY**

<u>Mary S. Armstrong /s/</u>	<u>2/11/99</u>
Mary S. Armstrong, Director of Development	Date
EOSDIS Core System Project	

**Raytheon Systems Company**  
Upper Marlboro, Maryland

This page intentionally left blank.

# Preface

---

This document is a contract deliverable with an approval code 2. As such, it does not require formal Government approval, however, the Government reserves the right to request changes within 45 days of the initial submittal. Once approved, contractor changes to this document are handled in accordance with Class I and Class II change control requirements described in the EOS Configuration Management Plan, and changes to this document shall be made by document change notice (DCN) or by complete revision. The ECS configuration control board has reviewed it, and this final version is being delivered prior to the Release Readiness Review.

This document is a final delivery of CDRL 305, Release 4 Segment/Design Specification of the ECS Project (305-CD-100-005).

Any questions may be addressed to:

Data Management Office  
The ECS Project Office  
Raytheon Systems Company  
1616 McCormick Drive  
Upper Marlboro, MD 20774-5301

This page intentionally left blank.

# Abstract

---

The Release 4 Segment/Design Specification is an overview description of the ECS Project. The functionality of the ECS software is described at the subsystem, Computer Software Configuration Item (CSCI), Computer Software Component (CSC), and process levels. Architecture and context diagrams illustrate the process interconnections within the ECS CSCIs and the external connections to other CSCIs, subsystems, and specified segment interfaces. Interface description tables describe the data, messages, notifications, or status information that occurs at each level of functionality within the ECS. A basic description of the Commercial Off The Shelf (COTS) software and hardware used in ECS is included. Also, a more detailed class level of documentation is offered from the output of the on-line tool ABC++.

Detailed design in this document is the level of information derived from requirement sources, and used by the development team to complete the ECS design implementation for a software system at a 4PX state of maturity.

**Keywords:** Release 4, Overview, SDPS, CSMS, Design, Detailed Design, Subsystem, Architecture, Software, Hardware, Object Oriented, Security, Gateway, System Management, Reports, User Interface, GUI.

This page intentionally left blank.

# Change Information Page

<b>List of Effective Pages</b>			
<b>Page Number</b>	<b>Issue</b>		
Title	Submitted as Final		
iii through xx	Submitted as Final		
1-1 and 1-2	Submitted as Final		
2-1 through 2-8	Submitted as Final		
3-1 through 3-10	Submitted as Final		
4-1 through 4-302	Submitted as Final		
5-1 through 5-4	Submitted as Final		
A-1 through A-268	Submitted as Final		
B-1 through B-142	Submitted as Final		
AB-1 through AB-10	Submitted as Final		
<b>Document History</b>			
<b>Document Number</b>	<b>Status/Issue</b>	<b>Publication Date</b>	<b>CCR Number</b>
305-CD-100-001	Draft	January 1998	97-1774
305-CD-100-001	Updated Draft	March 1998	98-0303
305-CD-100-002	Draft	May 1998	98-0630
305-CD-100-003	Draft	July 1998	98-0862
305-CD-100-004	Draft	December 1998	98-1208
305-CD-100-005	Submitted as Final	February 1999	99-0113

This page intentionally left blank.

# Contents

---

## Preface

## Abstract

### 1. Introduction

1.1	Purpose and Scope .....	1-1
1.2	Document Organization .....	1-2

### 2. Related Documentation

2.1	Parent Documents .....	2-1
2.2	Applicable Documents .....	2-1
2.3	Information Documents Not Referenced .....	2-8

### 3. System Description

3.1	Release 4 Objectives .....	3-1
3.1.1	Release 4 Capabilities .....	3-1
3.1.2	Release 4 Instrument and DAAC Support .....	3-2
3.2	Release 4 Architecture Overview .....	3-3
3.2.1	Release 4 Context Description .....	3-6
3.2.2	Release 4 Architecture.....	3-7

### 4. Subsystem Description

4.1	Data Server Subsystem Overview .....	4-3
4.1.1	Science Data Server Software Description .....	4-8
4.1.2	Data Distribution Software Description .....	4-23
4.1.3	Storage Management Software Description.....	4-31

4.1.4	Data Server Subsystem Hardware .....	4-45
4.2	Ingest Subsystem Overview .....	4-50
4.2.1	INGST Software Description .....	4-54
4.2.2	Ingest Subsystem Hardware .....	4-66
4.3	Client Subsystem Overview .....	4-69
4.3.1	Desktop Software Description .....	4-73
4.3.2	Workbench Software Description .....	4-80
4.4	Data Management Subsystem Overview .....	4-87
4.4.1	Data Dictionary Software Description .....	4-90
4.4.2	Information Manager Software Description.....	4-98
4.4.3	V0 Gateway Software Description.....	4-104
4.4.4	Data Management Subsystem Hardware .....	4-111
4.5	Interoperability Subsystem Overview .....	4-113
4.5.1	Advertising Service Software Description.....	4-116
4.5.2	Interoperability Subsystem Hardware Components.....	4-124
4.6	Planning Subsystem Overview.....	4-125
4.6.1	Production Planning (PLANG) Software Description .....	4-128
4.6.2	Planning Subsystem Hardware Components .....	4-135
4.7	Data Processing Subsystem Overview .....	4-137
4.7.1	Processing Software Description .....	4-141
4.7.2	Algorithm Integration and Test Tools Software Description.....	4-154
4.7.3	Data Processing Hardware Components .....	4-163
4.8	Communications Subsystem Overview .....	4-169
4.8.1	Subscription Server Software Description .....	4-175
4.8.2	ASTER DAR Gateway Software Description.....	4-180
4.8.3	E-mail Parser Gateway Software Description.....	4-185
4.8.4	Landsat 7 Gateway Server Software Description.....	4-189
4.8.5	Distributed Computing Environment Support Group Description .....	4-192
4.8.6	Remote File Access Group - File Transfer Protocol Description .....	4-196
4.8.7	Remote File Access Group - File Transfer Protocol Notification.....	4-199
4.8.8	Remote File Access Group - Bulk Data Server Description .....	4-201
4.8.9	Remote File Access Group - Network File System Description.....	4-205
4.8.10	Remote File Access Group - Filecopy Description.....	4-207

4.8.11	Mail Support Group Description.....	4-209
4.8.12	Virtual Terminal Description .....	4-213
4.8.13	Cryptographic Management Interface Software Description.....	4-216
4.8.14	Domain Name Server Software Description .....	4-218
4.8.15	Infrastructure Libraries Group Description .....	4-221
4.9	Systems Management Subsystem Overview.....	4-226
4.9.1	MCI Software Description - Network and Enterprise Management Component .....	4-232
4.9.2	MCI - Security Functional Component.....	4-239
4.9.3	MCI - Accountability Management Component.....	4-243
4.9.4	MCI - Trouble Ticket Component .....	4-253
4.9.5	MCI - Network Backup/Restore Component.....	4-259
4.9.6	MCI - ASTER E-mail Handler Component.....	4-261
4.9.7	MACI Software Description .....	4-266
4.9.8	MLCI - Baseline Manager Component.....	4-272
4.9.9	MLCI - Inventory/Logistics/Maintenance Manager Component.....	4-276
4.9.10	MLCI - Software Change Manager Component .....	4-279
4.9.11	MLCI - Change Request Manager Component.....	4-281
4.9.12	MLCI - Software Distribution Manager Component.....	4-283
4.9.13	MLCI - Software License Manager Component.....	4-284
4.10	Internetworking Subsystem (ISS) Overview.....	4-294
4.10.1	Internetworking Subsystem Description .....	4-295
4.10.2	Release 4 Network COTS Hardware.....	4-299

## **5. Limitations of Current Implementation**

### **List of Figures**

4.1-1.	Data Server Subsystem Context Diagram.....	4-4
4.1.1.2-1.	SDSRV CSCI Context Diagram .....	4-9
4.1.1.3-1.	SDSRV CSCI Architecture Diagram.....	4-12
4.1.2.2-1.	DDIST CSCI Context Diagram .....	4-24
4.1.2.3-1.	DDIST CSCI Architecture Diagram.....	4-26

4.1.3.2-1. STMGT CSCI Context Diagram .....	4-32
4.1.3.3-1. STMGT CSCI Architecture Diagram .....	4-35
4.2-1. Ingest Subsystem Context Diagram.....	4-51
4.2.1.2-1. INGST CSCI Context Diagram .....	4-56
4.2.1.3-1. INGST CSCI Architecture Diagram.....	4-58
4.3-1. Client Subsystem Context Diagram.....	4-69
4.3.1.2-1. DESKT CSCI Context Diagram .....	4-74
4.3.1.3-1. DESKT CSCI Architecture Diagram.....	4-76
4.3.2.2.-1. WKBCH CSCI Context Diagram .....	4-81
4.3.2.3-1. EOSView Architecture Diagram .....	4-82
4.3.2.3-2. DAR Tool Architecture Diagram.....	4-83
4.3.2.3-3. URT Architecture Diagram.....	4-84
4.4-1. Data Management Subsystem Context Diagram .....	4-87
4.4.1.2-1. Data Dictionary CSCI Context Diagram .....	4-91
4.4.1.3-1. Data Dictionary CSCI Architecture Diagram .....	4-93
4.4.2.2-1. Local Information Manager CSCI Context Diagram.....	4-99
4.4.2.3-1. Local Information Manager CSCI Architecture Diagram .....	4-101
4.4.3.2-1. V0 GTWAY CSCI Context Diagram .....	4-105
4.4.3.3-1. V0 GTWAY CSCI Architecture Diagram.....	4-107
4.5-1. Interoperability Subsystem Context Diagram.....	4-114
4.5.1.2-1. Advertising Service CSCI Context Diagram .....	4-117
4.5.1.3-1. Advertising Service Architecture Diagram.....	4-119
4.6-1. Planning Subsystem Context Diagram .....	4-125
4.6.1.2-1. PLANG CSCI Context Diagram.....	4-128
4.6.1.3-1. PLANG CSCI Architecture Diagram .....	4-130
4.7-1. Data Processing Subsystem Context Diagram.....	4-138
4.7.1.2-1. PRONG CSCI Context Diagram .....	4-142
4.7.1.3-1. PRONG CSCI Architecture Diagram .....	4-145
4.7.2.2-1. AITTL Context Diagram .....	4-155

4.7.2.3-1. AITTL CSCI Architecture Diagram .....	4-157
4.8-1. CSS Subsystem Context Diagram .....	4-169
4.8.1.2-1. Subscription Server Context Diagram .....	4-175
4.8.1.3-1. Subscription Server Architecture Diagram .....	4-177
4.8.2.2-1. ASTER DAR Gateway Context Diagram.....	4-181
4.8.2.3-1. ASTER DAR Gateway Architecture Diagram .....	4-182
4.8.3.2-1. E-mail Parser Gateway Context Diagram.....	4-185
4.8.3.3-1. E-mail Parser Gateway Architecture diagram.....	4-187
4.8.4.2-1. Landsat 7 Gateway Server Context Diagram.....	4-190
4.8.4.3-1. Landsat 7 Gateway Server Architecture Diagram .....	4-190
4.8.5.2-1. Distributed Computing Environment Context Diagram .....	4-193
4.8.5.3-1. Distributed Computing Environment Architecture Diagram.....	4-194
4.8.6.2-1. File Transfer Protocol Server Context Diagram .....	4-197
4.8.6.3-1. File Transfer Protocol Architecture Diagram .....	4-197
4.8.7.2-1. File Transfer Protocol Notification Context Diagram .....	4-199
4.8.7.3-1. File Transfer Protocol Notification Architecture Diagram .....	4-200
4.8.8.1-1. Bulk Data Server Protocol compared with ONC Protocols.....	4-202
4.8.8.2-1. Bulk Data Server Context Diagram .....	4-203
4.8.8.3-1. Bulk Data Server Architecture Diagram .....	4-204
4.8.9.2-1. Network File System Context Diagram.....	4-205
4.8.9.3-1. Network File System Architecture Diagram.....	4-206
4.8.10.2-1. Filecopy Context Diagram.....	4-207
4.8.10.3-1. Filecopy Architecture Diagram.....	4-208
4.8.11.2-1. E-mail and Bulletin Board Servers Context Diagram.....	4-210
4.8.11.3-1. E-mail and Bulletin Board Servers Architecture Diagram .....	4-211
4.8.12.2-1. Virtual Terminal Context Diagram.....	4-213
4.8.12.3-1. Virtual Terminal Architecture Diagram.....	4-214
4.8.13.2-1. Cryptographic Management Interface Context Diagram.....	4-216
4.8.14.1-1. Domains Hierarchy Diagram .....	4-218

4.8.14.1-2. DNS Domains of the ECS Project Diagram .....	4-219
4.8.14.1-3. ECS Topology Domains Diagram .....	4-219
4.8.14.2-1. Domain Name Server Context Diagram .....	4-220
4.9-1. System Management Subsystem Context Diagram.....	4-226
4.9.1.2-1. Network and Enterprise Management Framework Context Diagram.....	4-234
4.9.1.3-1. Network and Enterprise Management Framework Architecture Diagram .....	4-235
4.9.2.2-1. Security Service CSC Context Diagram .....	4-239
4.9.2.3-1. Security Service CSC Architecture Diagram.....	4-240
4.9.3.2-1. Accountability Management Service Context Diagram .....	4-245
4.9.3.3-1. Accountability Management Service Architecture Diagram .....	4-247
4.9.4.2-1. Trouble Ticket Context Diagram .....	4-253
4.9.4.3-1. Trouble Ticket Architecture Diagram.....	4-255
4.9.5.3-1. Backup/Restore Architecture Diagram .....	4-260
4.9.6.2-1. ASTER E-mail Header Handler Context Diagram .....	4-262
4.9.6.3-1. ASTER E-mail Header Handler Architecture Diagram.....	4-263
4.9.7.2-1. MACI CSCI Context Diagram.....	4-267
4.9.7.3-1. MACI CSCI Architecture Diagram .....	4-269
4.9.8.2-1. Baseline Manager Context Diagram .....	4-273
4.9.8.3-1. Baseline Manager Architecture Diagram.....	4-273
4.9.9.3-1. ILM Manager Architecture Diagram .....	4-277
4.9.13.2-1. Software License Manager Context Diagram .....	4-286
4.9.13.3-1. Software License Manager Architecture Diagram.....	4-287
4.10.1.1-1. Release 4 DAAC Networks: Generic Architecture Diagram.....	4-296
4.10.1.2-1. SMC Network Architecture Diagram .....	4-297

## List of Tables

4.1-1. Data Server Subsystem Interface Events .....	4-4
4.1.1.2-1. SDSRV CSCI Interface Events.....	4-9
4.1.1.4-1. SDSRV CSCI Processes .....	4-13

4.1.1.5-1. SDSRV CSCI Process Interface Events .....	4-14
4.1.1.6-1. SDSRV CSCI Data Stores .....	4-23
4.1.2.2-1. DDIST CSCI Interface Events .....	4-24
4.1.2.4-1. DDIST CSCI Processes .....	4-26
4.1.2.5-1. DDIST CSCI Process Interface Events.....	4-27
4.1.2.6-1. DDIST CSCI Data Stores .....	4-31
4.1.3.2-1. STMGT CSCI Interface Events .....	4-33
4.1.3.4-1. STMGT CSCI Processes.....	4-36
4.1.3.5-1. STMGT CSCI Process Interface Events.....	4-37
4.1.3.6-1. STMGT CSCI Data Stores.....	4-43
4.1.4.1-1. Access Control and Management HWCI Descriptions.....	4-46
4.1.4.2-1. Data Repository HWCI Descriptions.....	4-48
4.1.4.3-1. Working Storage HWCI Descriptions .....	4-48
4.2-1. Ingest Subsystem Interface Events .....	4-51
4.2.1.2-1. INGST CSCI Interface Events.....	4-56
4.2.1.4-1. INGST CSCI Processes .....	4-59
4.2.1.5-1. INGST CSCI Process Interface Events.....	4-60
4.2.1.6-1. INGST CSCI Data Stores .....	4-65
4.3-1. Client Subsystem Interface Events .....	4-70
4.3.1.2-1. DESKT CSCI Interface Events.....	4-75
4.3.1.4-1. DESKT CSCI Processes .....	4-77
4.3.1.5-1. DESKT CSCI Process Interface Events.....	4-78
4.3.1.6-1. DESKT CSCI Data Stores .....	4-80
4.3.2.2-1. WKBCH CSCI Interface Events.....	4-81
4.3.2.4-1. WKBCH CSCI Processes .....	4-84
4.3.2.5-1. WKBCH CSCI Process Interface Events.....	4-85
4.3.2.6-1. WKBCH CSCI Data Stores .....	4-86
4.4-1. Data Management Subsystem Interface Events.....	4-88
4.4.1.2-1. Data Dictionary CSCI Interface Events .....	4-91

4.4.1.4-1. Data Dictionary CSCI Processes .....	4-94
4.4.1.5-1. Data Dictionary CSCI Process Interface Events .....	4-95
4.4.1.6-1. Data Dictionary CSCI Data Stores .....	4-98
4.4.2.2-1. Local Information Manager CSCI Interface Events .....	4-99
4.4.2.4-1. Local Information Manager CSCI Process .....	4-101
4.4.2.5-1. Local Information Manager Process Interface Events .....	4-102
4.4.2.6-1. Local Information Manager Data Stores.....	4-104
4.4.3.2-1. V0 GTWAY CSCI Interface Events .....	4-105
4.4.3.4-1. V0 GTWAY CSCI Processes .....	4-108
4.4.3.5-1. V0 GTWAY CSCI Process Interface Events .....	4-109
4.4.3.6-1. V0 GTWAY CSCI Data Store.....	4-111
4.5-1. Interoperability Subsystem Interface Events .....	4-114
4.5.1.2-1. Advertising Service CSCI Interface Events .....	4-117
4.5.1.4-1. Advertising Service Processes .....	4-120
4.5.1.5-1. Advertising Service Process Interface Events .....	4-121
4.5.1.6-1. Advertising Service Data Stores .....	4-124
4.6-1. Planning Subsystem Interface Events.....	4-126
4.6.1.2-1. PLANG CSCI Interface Events .....	4-129
4.6.1.4-1. PLANG CSCI Processes .....	4-131
4.6.1.5-1. PLANG CSCI Process Interface Events .....	4-132
4.6.1.6-1. PLANG CSCI Data Stores.....	4-135
4.7-1. Data Processing Subsystem Interface Events .....	4-138
4.7.1.2-1. PRONG CSCI Interface Events .....	4-142
4.7.1.4-1. PRONG CSCI Processes .....	4-146
4.7.1.5-1. PRONG CSCI Process Interface Events.....	4-149
4.7.1.6-1. PRONG CSCI Data Stores.....	4-153
4.7.2.2-1. AITTL Interface Events .....	4-155
4.7.2.4-1. AITTL Processes .....	4-158
4.7.2.5-1. AITTL Process Interface Events .....	4-159

4.7.2.6-1. AITTTL Data Stores .....	4-162
4.8-1. CSS Subsystem Interface Events .....	4-170
4.8.1.2-1. Subscription Server Interface Events .....	4-176
4.8.1.4-1. Subscription Server Processes .....	4-177
4.8.1.5-1. Subscription Server Process Interfaces .....	4-178
4.8.1.6-1. Subscription Server Data Stores .....	4-180
4.8.2.2-1. ASTER DAR Gateway Interface Events .....	4-182
4.8.2.4-1. ASTER DAR Gateway Processes.....	4-183
4.8.2.5-1. ASTER DAR Interface Events .....	4-184
4.8.3.2-1. E-mail Parser Gateway Interface Events.....	4-186
4.8.3.4-1. E-mail Parser Gateway Processes .....	4-187
4.8.3.5-1. E-mail Parser Gateway Process Interface Events .....	4-188
4.8.4.2-1. Landsat 7 Gateway Server Interface Events .....	4-190
4.8.4.4-1. Landsat 7 Gateway Server Process .....	4-191
4.8.4.5-1. Landsat 7 Gateway Server Interface Events .....	4-191
4.8.5.2-1. Distributed Computing Environment Interface Events .....	4-193
4.8.5.4-1. Distributed Computing Environment Processes .....	4-195
4.8.5.5-1. Distributed Computing Environment Interface Events .....	4-195
4.8.5.6-1. Distributed Computing Environment Data Stores .....	4-196
4.8.6.2-1. File Transfer Protocol Interface Events .....	4-197
4.8.6.4-1. File Transfer Protocol Processes.....	4-198
4.8.6.5-1. File Transfer Protocol Interface Events .....	4-198
4.8.7.2-1. File Transfer Protocol Notification Interface Events.....	4-199
4.8.7.4-1. File Transfer Protocol Notification Processes .....	4-200
4.8.7.5-1. File Transfer Protocol Notification Interfaces .....	4-201
4.8.7.6-1. File Transfer Protocol Data Stores.....	4-201
4.8.8.2-1. Bulk Data Server Interface Events.....	4-203
4.8.8.4-1. Bulk Data Server Processes .....	4-204
4.8.8.5-1. Bulk Data Server Interface Events.....	4-205

4.8.9.2-1. Network File System Interface Events.....	4-206
4.8.9.4-1. Network File System Processes.....	4-206
4.8.9.5-1. Network File System Interface Events.....	4-207
4.8.10.2-1. Filecopy Interface Events.....	4-208
4.8.10.4-1. Filecopy Process .....	4-208
4.8.10.5-1. Filecopy Interface Events.....	4-209
4.8.11.2-1. E-mail and Bulletin Board Servers Interface Events .....	4-210
4.8.11.4-1. E-mail and Bulletin Board Servers Processes .....	4-211
4.8.11.5-1. E-mail and Bulletin Board Interface Events .....	4-212
4.8.12.2-1. Virtual Terminal Interface Events.....	4-214
4.8.12.4-1. Virtual Terminal Processes .....	4-214
4.8.12.5-1. Virtual Terminal Interface Events.....	4-215
4.8.13.2-1. Cryptographic Management Interface Events.....	4-216
4.8.13.4-1. Cryptographic Management Interface Processes.....	4-217
4.8.13.5-1. Cryptographic Management Interface Interface Events .....	4-217
4.8.13.6-1. Cryptographic Management Interface Data Stores.....	4-217
4.8.14.4-1. Domain Name Server Processes .....	4-220
4.8.14.5-1. Domain Name Process Interface Events.....	4-221
4.8.14.6-1. Domain Name Server Data Stores .....	4-221
4.8.15.1-1. Infrastructure Libraries .....	4-221
4.8.15.5-1. Infrastructure Library Group Interfaces .....	4-222
4.9-1. System Management Subsystem Interface Events .....	4-227
4.9.1.2-1. Management Framework Interface Events .....	4-235
4.9.1.4-1. Network and Enterprise Management Framework Processes .....	4-236
4.9.1.5-1. Network and Enterprise Management Framework Process Interfaces .....	4-237
4.9.1.6-1. Network and Enterprise Management Framework Data Stores .....	4-238
4.9.2.2-1. Security Service CSC Interface Events.....	4-240
4.9.2.4-1. Security Service CSC Processes .....	4-241
4.9.2.5-1. Security Process Interface Events .....	4-241

4.9.2.6-1. Security Data Stores.....	4-243
4.9.3.2-1. Accountability Management Service Interface Events .....	4-245
4.9.3.4-1. Accountability Management Processes .....	4-248
4.9.3.5-1. Accountability Management Service Process Interface Events .....	4-250
4.9.3.6-1. Accountability Management Data Stores .....	4-252
4.9.4.2-1. Trouble Ticket Interface Events.....	4-254
4.9.4.4-1. Trouble Ticket Processes .....	4-256
4.9.4.5-1. Trouble Ticket Process Interface Events .....	4-257
4.9.4.6-1. Trouble Ticket Data Stores .....	4-259
4.9.5.4-1. Backup/Restore Processes .....	4-260
4.9.5.5-1. Backup/Restore Process Interface Events.....	4-261
4.9.5.6-1. Backup/Restore Data Stores .....	4-261
4.9.6.2-1. ASTER E-mail Header Handler Interface Events.....	4-263
4.9.6.4-1. ASTER E-mail Header Handler Processes .....	4-264
4.9.6.5-1. ASTER E-mail Header Handler Process Interface Events .....	4-264
4.9.6.6-1. ASTER E-mail Header Handler Data Stores .....	4-266
4.9.7.2-1. MACI CSCI Interface Events .....	4-268
4.9.7.4-1. MACI CSCI Processes.....	4-269
4.9.7.5-1. MACI CSCI Process Interface Events .....	4-270
4.9.8.4-1. Baseline Manager Processes .....	4-274
4.9.8.5-1. Baseline Manager Process Interface Events .....	4-274
4.9.8.6-1. Baseline Manager Data Stores .....	4-275
4.9.9.4-1. ILM Manager Processes.....	4-278
4.9.9.5-1. ILM Manager Process Interface Events.....	4-278
4.9.9.6-1. ILM Manager Data Stores.....	4-279
4.9.10.4-1. Software Change Manager Processes .....	4-280
4.9.10.6-1. Software Change Manager Data Stores .....	4-280
4.9.11.4-1. Change Request Manager Processes.....	4-282
4.9.11.6-1. Change Request Manager Data Stores.....	4-283

4.9.12.4-1. Software Distribution Manager Processes .....	4-284
4.9.12.6-1. Software Distribution Manager Data Stores .....	4-284
4.9.13.4-1. Software License Manager Processes .....	4-287
4.9.13.5-1. Software License Manager Interface Events.....	4-289
4.9.13.6-1. Software License Manager Data Stores .....	4-290
4.10.1.5-1. Internetworking Subsystem Baseline Documentation List.....	4-299
4.10.2-1. Release 4 Networking Hardware for ECS Networks .....	4-299

## **Appendix A. Process**

Subsystem DSS .....	1
Subsystem INS .....	43
Subsystem CLS .....	58
Subsystem DMS .....	74
Subsystem IOS .....	89
Subsystem PLS.....	128
Subsystem DPS .....	172
Subsystem CSS .....	232
Subsystem MSS .....	248

## **Appendix B. Libraries**

### **Abbreviations and Acronyms**

# 1. Introduction

---

## 1.1 Purpose and Scope

The purpose of the Segment/Design Specification for the Earth Observing System (EOS) Data and Information System (EOSDIS) Core System (ECS) is to provide an overview of the hardware and software subsystems of the project. This document describes the detailed design of each ECS software subsystem implemented to satisfy the allocated and derived functional and performance requirements. This document also provides basic descriptions of the Commercial Off The Shelf (COTS) hardware and software used in the ECS. This document contains:

- Functional overviews of each Computer Software Configuration Item (CSCI)
- Context diagrams of each CSCI
- Interface event descriptions based on the context diagrams
- Process architecture diagrams
- Interface description tables based on the architecture diagrams
- CSCI data stores (databases as they relate to the process architecture)
- CSCI functions allocated to processes. For data servers, this includes descriptions of the functionality offered to clients via the server interfaces. For Graphical User Interface (GUI) applications, it describes the functionality provided to the GUI users.
- Specific limitations of the capabilities provided
- Summary of object classes listed by CSCI
- Summary of class libraries listed by CSCI
- Abbreviations and Acronyms

Hyper-linked on-line documentation generated by ABC++ is provided to accompany this document. The various subsystems, subdirectories, processes, libraries, and classes can be browsed down to the code level. This documentation can be accessed through the Universal Resource Locator (URL) <http://ecsdocs.east.hitc.com:88> for those who have access to the ECS internally at the Landover Facility. For anyone not having access to the ECS internally at the Landover facility, the Data Management Office of the ECS Project must be contacted for access.

Under every process or library documented in Appendix A or B, there is a directory string. This string indicates the location of the Makefile and sources for the library.

This same directory string can be used to locate on-line documentation. This is accomplished by replacing the “/ecs/formal” part of the directory string with the URL for the on-line documentation associated with the page on which the classes are listed along with the names of processes and libraries which share those classes.

## **1.2 Document Organization**

The remainder of this document is organized as follows:

- Section 2: Related Documentation
- Section 3: System Description
- Section 4: Subsystem Descriptions
- Section 5: Limitations of Current Implementation
- Appendix A: Software Processes
- Appendix B: Software Libraries
- Abbreviations and Acronyms

## 2. Related Documentation

---

### 2.1 Parent Documents

The parent documents are the documents from which the scope and content of this Design Specification are derived. These documents are listed below.

194-207-SEI	System Design Specification for the ECS Project
305/DV2 (F)	Segment/Design Specifications
212-WP-002	Game Plan (preliminary copy) for the ECS Project

### 2.2 Applicable Documents

Documents referenced in this document are listed below.

#### *912-series Disk Partition-Common*

912-TDA-002	Application Server
912-TDE-012	EDC MSS / MSS Backup Server
912-TDG-012	GSFC MSS / MSS Backup Server
912-TDL-012	LaRC MSS / MSS Backup Server
912-TDS-012	SMC MSS / MSS Backup Server
912-TDV-112	VATC-SMC MSS / MSS Backup Server

#### *920-series General documents*

920-TDE-001	EDC Hardware Diagram (Design)
920-TDG-001	GSFC Hardware Diagram (Design)
920-TDL-001	LaRC Hardware Diagram (Design)
920-TDS-001	SMC Hardware Diagram (Design)
920-TDV-101	VATC-SMC Hardware Diagram (Design)
920-TDE-002	EDC Hdwe/Sfwe Mapping (COTS)
920-TDG-002	GSFC Hdwe/Sfwe Mapping (COTS)
920-TDL-002	LaRC Hdwe/Sfwe Mapping (COTS)
920-TDS-002	SMC Hdwe/Sfwe Mapping (COTS)

920-TDV-102	VATC-SMC Hdwe/Sfwe Mapping (COTS)
920-TDE-003	EDC System Infrastructure Assignments (DCE/ DNS/DTS Infra.)
920-TDG-003	GSFC System Infrastructure Assignments (DCE/ DNS/DTS Infra.)
920-TDL-003	LaRC System Infrastructure Assignments (DCE/ DNS/DTS Infra.)
920-TDS-003	SMC System Infrastructure Assignments (DCE/ DNS/DTS Infra.)
920-TDV-103	VATC-SMC System Infrastructure Assignments (DCE/DNS/DTS Infra.)
920-TDE-004	EDC Floor Plan
920-TDG-004	GSFC Floor Plan
920-TDL-004	LaRC Floor Plan
920-TDS-004	SMC Floor Plan
920-TDV-104	VATC-SMC Floor Plan
920-TDE-005	EDC Cable Management Plan
920-TDG-005	GSFC Cable Management Plan
920-TDL-005	LaRC Cable Management Plan
920-TDS-005	SMC Cable Management Plan
920-TDV-105	VATC-SMC Cable Management Plan
920-TDE-007	EDC Vendor Documentation List
920-TDG-007	GSFC Vendor Documentation List
920-TDL-007	LaRC Vendor Documentation List
920-TDE-008	EDC Mount Points
920-TDG-008	GSFC Mount Points
920-TDL-008	LaRC Mount Points
920-TDS-008	SMC Mount Points
920-TDV-108	VATC-SMC Mount Points
920-TDE-009	EDC DAAC Baseline Hardware / Database Mapping
920-TDG-009	GSFC DAAC Baseline Hardware / Database Mapping
920-TDL-009	LaRC DAAC Baseline Hardware / Database Mapping
920-TDS-009	SMC DAAC Baseline Hardware / Database Mapping

920-TDV-109	VATC-SMC DAAC Baseline Hardware / Database Mapping
920-TDE-010	EDC DAAC Baseline Database Configuration Listing
920-TDG-010	GSFC DAAC Baseline Database Configuration Listing
920-TDL-010	LaRC DAAC Baseline Database Configuration Listing
920-TDS-010	SMC DAAC Baseline Database Configuration Listing
920-TDV-110	VATC-SMC DAAC Baseline Database Configuration Listing
920-TDE-011	EDC DAAC Baseline Sybase Logging Mapping
920-TDG-011	GSFC DAAC Baseline Sybase Logging Mapping
920-TDL-011	LaRC DAAC Baseline Sybase Logging Mapping
920-TDS-011	SMC DAAC Baseline Sybase Logging Mapping
920-TDV-111	VATC-SMC DAAC Baseline Sybase Logging Mapping
920-TDE-012	EDC DAAC SCSI Cable Management Plan
920-TDG-012	GSFC DAAC SCSI Cable Management Plan
920-TDL-012	LaRC DAAC SCSI Cable Management Plan

***921-series Network Infrastructure documents***

921-TDE-001	EDC Network Overview Diagram (LAN Topology)
921-TDG-001	GSFC Network Overview Diagram (LAN Topology)
921-TDL-001	LaRC Network Overview Diagram (LAN Topology)
921-TDS-001	SMC Network Overview Diagram (LAN Topology)
921-TDV-101	VATC-SMC Network Overview Diagram (LAN Topology)
921-TDE-002	EDC Hardware / Network Diagram
921-TDG-002	GSFC Hardware / Network Diagram
921-TDL-002	LaRC Hardware / Network Diagram
921-TDS-002	SMC Hardware / Network Diagram
921-TDV-102	VATC-SMC Hardware / Network Diagram
921-TDE-003	EDC IP Address Assignments (DAAC Hosts)
921-TDG-003	GSFC IP Address Assignments (DAAC Hosts)
921-TDL-003	LaRC IP Address Assignments (DAAC Hosts)
921-TDS-003	SMC IP Address Assignments (DAAC Hosts)

921-TDV-103	VATC-SMC IP Address Assignments (DAAC Hosts)
921-TDE-004	EDC IP Address Assignments (DAAC Network Hardware)
921-TDG-004	GSFC IP Address Assignments (DAAC Network Hardware)
921-TDL-004	LaRC IP Address Assignments (DAAC Network Hardware)
921-TDS-004	SMC IP Address Assignments (DAAC Network Hardware)
921-TDV-104	VATC-SMC IP Address Assignments (DAAC Network Hardware)
921-TDE-005	EDC Dual-Homed Host Static Routes
921-TDG-005	GSFC Dual-Homed Host Static Routes
921-TDL-005	LaRC Dual-Homed Host Static Routes
921-TDE-006	EDC Ingest Host Static Routes
921-TDG-006	GSFC Ingest Host Static Routes
921-TDL-006	LaRC Ingest Host Static Routes

***922-series Disk Partitions documents***

922-TDE-001	EDC APC Server
922-TDG-001	GSFC APC Server
922-TDL-001	LaRC APC Server
922-TDS-001	SMC APC Server
922-TDV-101	VATC-SMC APC Server
922-TDE-002	EDC Application Server
922-TDG-002	GSFC Application Server
922-TDL-002	LaRC Application Server
922-TDS-002	SMC Application Server
922-TDV-102	VATC-SMC Application Server
922-TDG-003	GSFC AQA Host
922-TDL-003	LaRC AQA Host
922-TDS-004	SMC Bulletin Board Server
922-TDV-104	VATC-SMC Bulletin Board Server
922-TDE-005	EDC CSS Server
922-TDG-005	GSFC CSS Server

922-TDL-005	LaRC CSS Server
922-TDS-005	SMC CSS Server
922-TDV-105	VATC-SMC CSS Server
922-TDE-006	EDC DIP Server
922-TDG-006	GSFC DIP Server
922-TDL-006	LaRC DIP Server
922-TDS-006	SMC DIP Server
922-TDV-106	VATC-SMC DIP Server
922-TDE-007	EDC DRP Server
922-TDG-007	GSFC DRP Server
922-TDL-007	LaRC DRP Server
922-TDS-007	SMC DRP Server
922-TDV-107	VATC-SMC DRP Server
922-TDE-008	EDC ICL Server
922-TDG-008	GSFC ICL Server
922-TDL-008	LaRC ICL Server
922-TDS-008	SMC ICL Server
922-TDV-108	VATC-SMC ICL Server
922-TDE-009	EDC Interface Server
922-TDG-009	GSFC Interface Server
922-TDL-009	LaRC Interface Server
922-TDS-009	SMC Interface Server
922-TDV-109	VATC-SMC Interface Server
922-TDE-010	EDC MSS Server
922-TDG-010	GSFC MSS Server
922-TDL-010	LaRC MSS Server
922-TDD-010	SMC MSS Server
922-TDV-110	VATC-SMC MSS Server
922-TDE-011	EDC MSS File Server / CM Server

922-TDG-011	GSFC MSS File Server / CM Server
922-TDL-011	LaRC MSS File Server / CM Server
922-TDS-011	SMC MSS File Server / CM Server
922-TDV-111	VATC-SMC MSS File Server / CM Server
922-TDE-013	EDC Planning Server
922-TDG-013	GSFC Planning Server
922-TDL-013	LaRC Planning Server
922-TDS-013	SMC Planning Server
922-TDV-113	VATC-SMC Planning Server
922-TDE-014	EDC Queuing Server
922-TDG-014	GSFC Queuing Server
922-TDL-014	LaRC Queuing Server
922-TDS-014	SMC Queuing Server
922-TDV-114	VATC-SMC Queuing Server
922-TDE-015	EDC Science Processor
922-TDG-015	GSFC Science Processor
922-TDL-015	LaRC Science Processor
922-TDS-015	SMC Science Processor
922-TDV-115	VATC-SMC Science Processor
922-TDE-016	EDC SDSRV Server
922-TDG-016	GSFC SDSRV Server
922-TDL-016	LaRC SDSRV Server
922-TDS-016	SMC SDSRV Server
922-TDV-116	VATC-SMC SDSRV Server
922-TDE-017	EDC Sybase Staging Server
922-TDG-017	GSFC Sybase Staging Server
922-TDL-017	LaRC Sybase Staging Server
922-TDS-017	SMC Sybase Staging Server
922-TDV-117	VATC-SMC Sybase Staging Server

922-TDE-018	EDC WKS Server
922-TDG-018	GSFC WKS Server
922-TDL-018	LaRC WKS Server
922-TDS-018	SMC WKS Server
922-TDV-118	VATC-SMC WKS Server
922-TDE-019	EDC DIP Server #2
922-TDG-019	GSFC DIP Server #2
922-TDL-019	LaRC DIP Server #2
922-TDL-020	LaRC Science Processor # 2
922-TDE-021	EDC AIT Server
922-TDG-021	GSFC AIT Server
922-TDL-021	LaRC AIT Server
922-TDS-021	SMC AIT Server
922-TDV-121	VATC-SMC AIT Server
205-CD-004	Science User's Guide and Operations Procedures Handbook (Release B.0) for the ECS Project
311-CD-109	Subscription Server Database Design and Schema Specifications for the ECS Project
313-CD-006	Release 4 CSMS/SDPS Internal Interface Control Document for the ECS Project
333-CD-100	Release 4 SDP Toolkit User's Guide for the ECS Project
193-801-SD4	PGS Toolkit Requirements Specification for the ECS Project (a.k.a. GSFC 423-06-02)
193-WP-118	Algorithm Integration and Test Issues, White Paper for the ECS Project
194-WP-925	Science Software Integration and Test White Paper for the ECS Project
423-16-01	Data Production Software and Science Computing Facility (SCF) Standards and Guidelines
IMSV0-PD-SD-002 v1.0.14 950928	Messages and Development Data Dictionary for v5.0 IMS Client
COTS	XRP-II Manual

RFC 793 Transmission Control Protocol  
RFC 768 User Datagram Protocol  
RFC 791 Internet Protocol  
RFC 1597 Address Allocation for Private Internets  
WWW page <http://cmdm.east.hitc.com/baseline>

### **2.3 Information Documents Not Referenced**

The documents listed below, while not directly applicable, help in the maintenance of the delivered software.

423-41-02 Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System Core System

540-022 Goddard Space Flight Center, Earth Observing System Communications System Interface Requirements Document

560-EDOS-0211.0001 Goddard Space Flight Center, Interface Requirements Document between EDOS and the EOS Ground System

# 3. System Description

---

## 3.1 Release 4 Objectives

### 3.1.1 Release 4 Capabilities

The ECS capabilities are developed in increments called formal releases and identified by version numbers. Release 4, the initial version, controlled by Configuration Management, provides capabilities to support the ingest and archival of raw data obtained from the EOS AM Mission spacecraft 1, morning equator crossing spacecraft series (AM-1), and the Land Remote-Sensing Satellite (Landsat 7). Other capabilities provided by Release 4 include processing the data obtained, distributing raw or processed data as requested, quality assurance of processed data, supporting communication networks, and systems monitoring via interfaces with the ECS operations staff.

Release 4 capabilities include:

- ingest and archival of AM-1 and other science and engineering data from the EOS Data and Operations System (EDOS) and Landsat 7 Level 0 Reformatted (LOR) data from the Landsat 7 Processing System (LPS)
- ingest of Product Generation Executive (PGE) software from a Science Computing Facility (SCF) either electronically or via media tape
- archival, production, and Quality Assurance (QA) processing of AM-1 science data products, science software integration and test, and the associated communications network interfaces with the SCF
- operator interfaces for production and resource planning
- directory and inventory search, including a user browse capability via the Version Zero (V0) System user interface
- receiving requests for science data products and distributing science data products via FTP and 8mm tape
- interfacing with the Advanced Space-borne Thermal Emission and Reflection Radiometer (ASTER) Ground Data System (GDS) for the submission of Data Acquisition Requests (DARs) for data collection by the ASTER instrument aboard the AM-1 spacecraft
- ingest of ASTER Level 1A/1B data from the ASTER GDS
- managing the startup and shutdown of system network components, user registration and profile administration, database and archive administration, system data and file back-up and restores, system performance tuning and resource usage monitoring, and other routine operator duties

ECS Release 4 has been distributed to four site locations including: the System Management Center (SMC), located at the Goddard Space Flight Center (GSFC), the DAACs at GSFC, Langley Research Center (LaRC), and the Earth Resource Observation System (EROS) Data Center (EDC).

The ECS Release 4 communications network includes the National Aeronautics and Space Administration (NASA) Science Internet (NSI) and the EOSDIS Backbone Network (EBnet). These portions of the network are physically located at the SMC and at the DAAC sites. The communications network connects ECS to data providers at the EDOS, the Landsat 7 Processing System (LPS), the LaRC TRMM Information System (LaTIS), NOAA Affiliated Data Center (ADC), and the EOSDIS Version 0 system.

The data users for Release 4 are the science user community connected to the three DAACs, the SCFs, and the ASTER GDS.

### **3.1.2 Release 4 Instrument and DAAC Support**

ECS Release 4 supports the following instrument data operations and DAACs.

- **AM-1 Support:**

ECS Release 4 supports archiving, processing, and distributing instrument data from a complement of five AM-1 instruments. These instruments are ASTER, Clouds and Earth's Radiant Energy System (CERES), Multi-Imaging SpectroRadiometer (MISR), Moderate-Resolution Imaging SpectroRadiometer (MODIS), and Measurements Of Pollution In The Troposphere (MOPITT).

ECS Release 4 provides a communications network and data/information management support for:

- ASTER instrument data including the receipt of ASTER level 1 data on magnetic tape at EDC from Japan, and the production and distribution of higher level ASTER products by EDC.
- CERES instrument data including the receipt of CERES level 0 data and an interface with LaTIS to provide L0 data and ancillary data.
- MISR instrument data including the receipt of MISR level 0 data and the LaRC archive, production, and distribution of levels 1, 2 and 3 data and data products.
- MODIS instrument data including level 0 data: archive, production, and distribution of levels 1A and 1B Including distribution of the 1A and 1B data to the MODIS Data Processing System (MODAPS) for higher level processing.
- MOPITT instrument data including the receipt of MOPITT level 0 data, and the LaRC archive, production, and distribution of levels 1, 2 and 3 data. During the initial period after launch, the MOPITT higher level products are generated at the SCF.

- Landsat 7 Support:

ECS Release 4 provides a repository for Landsat 7 L0R (L0 reformatted) data. The ECS communications network includes interfaces with the Landsat 7 system elements, the Mission Management Office (MMO), and the Ground Data Processing System (GDPS). The GDPS is composed of the Landsat 7 Processing System and the Image Assessment System (IAS).

The ECS network interfaces are needed to receive product cost information, exchange of registration services, and system management status from the MMO. The ECS receives and stores Landsat 7 level 0R data, metadata, and browse data from the LPS. The ECS also receives calibration data and metadata from the IAS. ECS network interfaces for data search, data order, and data distribution services to Landsat 7 users are also provided in Release 4.

### 3.2 Release 4 Architecture Overview

The ECS Release 4 architecture comprises the logical items listed here. Commercial Off The Shelf (COTS) software and hardware is used, to the extent possible, to implement the ECS functionality of these logical items.

- System
- Segments
- Subsystems
- Computer software configuration items (CSCIs)
- Computer software components (CSCs)
- Processes

ECS Release 4 was built of the following two segments.

- CSMS – Communications and Systems Management Segment
- SDPS – Science Data Processing Segment

Each segment was in turn built of the following subsystems.

- CSMS:   CSS - Communications Subsystem  
          ISS - Internetworking Subsystem  
          MSS - System Management Subsystem
- SDPS:   CLS - Client Subsystem  
          DMS - Data Management Subsystem  
          IOS - Interoperability Subsystem

DSS - Data Server Subsystem

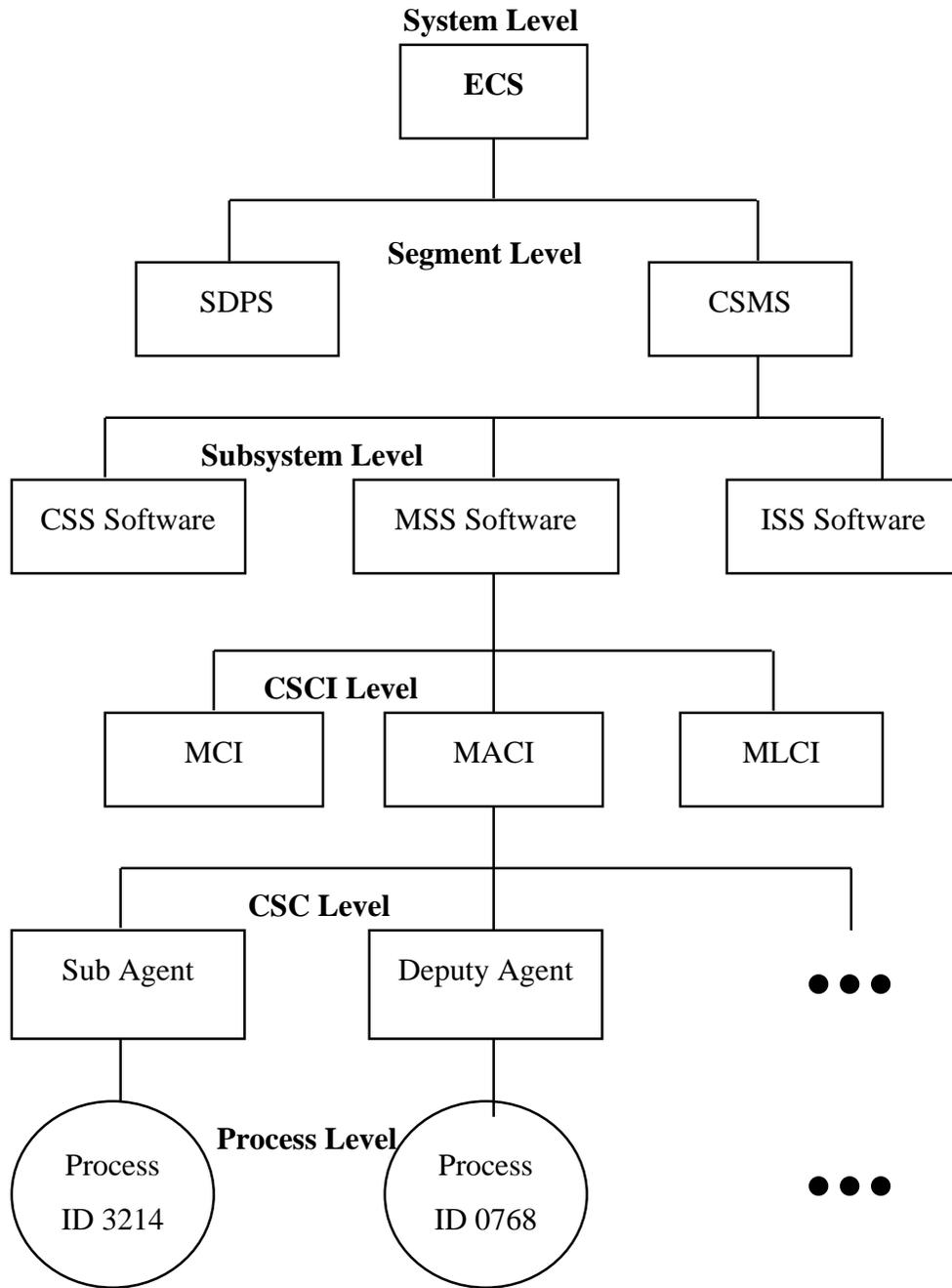
PLS - Planning Subsystem

DPS - Data Processing Subsystem

INS - Ingest Subsystem

- System:** A stand-alone composite of hardware, facilities, material, software, services, and personnel required for operation based upon a defined set of system level requirements.
- Segment:** A logical and functional set of related capabilities, implemented with COTS hardware and COTS and custom developed software to satisfy a defined set of system level requirements.
- Subsystem:** A logical set of related capabilities, implemented with COTS hardware and COTS and custom developed software to satisfy a defined subset of system level requirements.
- CSCI:** A logical set of related capabilities, implemented with COTS and custom developed software to satisfy a defined subset of system software requirements.
- CSC:** A logical set of related capabilities, implemented with COTS and custom developed software to satisfy a defined subset of software requirements.
- Process:** A logical set of software, written in a specific order and in a defined manageable size to manipulate data as part of a product-generating algorithm. A process can be separately compiled but is not always separately executable.

Figure 3.2-1 is a hierarchical software diagram. The hierarchical software diagram depicts an example of the decomposition levels used in the ECS design and described in this document. The diagram is also a graphical representation of the terms just described.

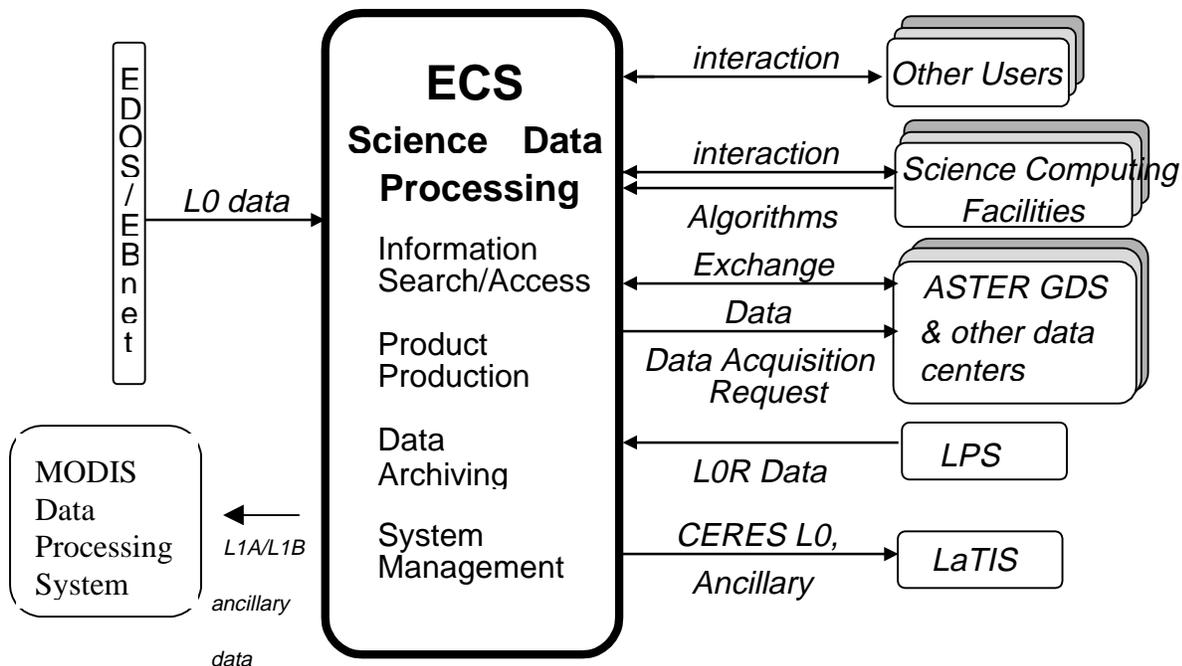


**Figure 3.2-1. Example Hierarchical Software Diagram**

### 3.2.1 Release 4 Context Description

ECS Release 4 provides the capability to collect and process satellite science data as depicted in Figure 3.2.1-1.

The Science Data Processing and Communications and Systems Management are the two segments of ECS Release 4 described in this document. The Science Data Processing Segment (SDPS) provides science data ingest and production, search and access functions, data archive, and system management capabilities. The SDPS receives AM-1 Level 0 science data from EDOS and LOR Data from the Landsat 7 Processing System. The SDPS exchanges data with the ASTER GDS and other affiliated data centers to obtain science and other data (i.e., engineering and ancillary) required for data production. It also connects with the ASTER GDS to submit ASTER Data Acquisition Requests (DARs) for the collection of science data by the ASTER instrument. Science algorithms, provided by the SCFs, are used in data production to transform data into higher level products (Level 1 to Level 4 products) for research. ECS uses SCF expertise to support the Quality Assurance activities of using the results of the Science Software Integration and Test (SSIT) activities to process data and verify the data production science algorithms prior to actual data production. The Communications and Systems Management Segment (CSMS) provides the communications infrastructure for ECS and systems management for all ECS hardware and software components. The CSMS provides the interconnection between users and service providers within ECS, transfer of information between subsystems, CSCIs, CSCs, and processes of ECS.



**Figure 3.2.1-1. Release 4 Context Diagram.**

The remaining sections of this document provide an overview of the ECS Release 4 design and as such do not deal specifically with the configuration of components at each EOSDIS site. For more information on the site unique configurations, refer to the 920-series of General documents. Each of the segments consists of subsystems as specified in Section 3.2.

### 3.2.2 Release 4 Architecture

#### 3.2.2.1 Subsystem Architecture

The ECS SDPS subsystems are depicted in Figure 3.2.2.1-1. A subsystem consists of the Commercial Off The Shelf (COTS) and/or ECS developed software and the COTS hardware needed for its execution. This document describes the software and hardware components of each subsystem. However, since the hardware configurations differ between the sites, the hardware descriptions in this document are at a generic level. Specific hardware and network configurations for each site are documented in the 921-series Network Infrastructure documents.

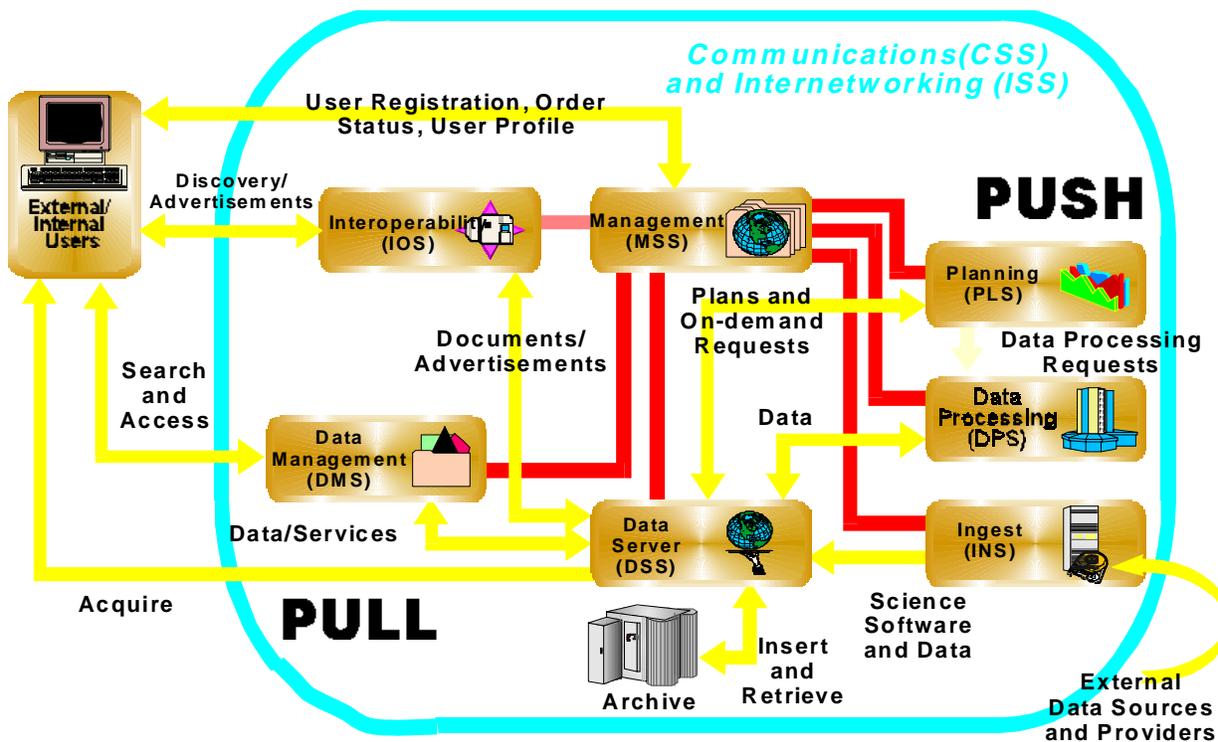


Figure 3.2.2.1-1. Subsystem Architecture Diagram

As shown in the subsystem architecture diagram, ECS Release 4 subsystems can be grouped into a 'Push' or 'Pull' category of functionality with the exception of DSS. The DMS and the IOS on the 'Pull' side of ECS also use the DSS functionality described on the 'Push' side of the ECS architecture. Data capture (ingest of data), storage management, planning and data processing of satellite or previously archived data from other sites make up the 'Push' side of the ECS architecture/design and consists of:

1. The DSS with the functions needed to
  - archive (insert) science data
  - search for and retrieve archived data
  - manage (create and delete items from) the archives
  - stage (store on physical media temporarily) data resources needed as input to data processing or resulting as output from their processing execution
2. The INS with interfaces to external suppliers (such as EDOS and LPS) to:
  - transfer data into ECS and temporarily store the data
  - provide staging capabilities for data waiting to be archived
  - provide operator interfaces for managing ingest operations
3. The DPS and PLS with:
  - DPS dispatching and monitoring the execution of the science software, and interfaces to DSS to stage the input data needed and archive the generated data
  - PLS providing long and short term planning of science data processing of the production environment and production resources

Information search and data retrieval makes up the 'Pull' side of the ECS architecture/design and consists of:

1. The CLS, DMS, and IOS with:
  - CLS providing user interfaces for data search and retrieval to science users and operators
  - DMS providing support for data search and retrieval across all ECS sites in conjunction with the DSS at each site. The DMS also provides a gateway as the interface to the Version 0 (V0) Information Management System (IMS) using the V0 IMS protocol
  - IOS managing advertisements of data and services it receives from the DSS or non-ECS users via the operations staff. IOS makes the advertisements available via a World Wide Web (WWW) interface for searching and browsing. Other ECS subsystems (i.e., CLS, PLS, DPS, DMS, and INS) use IOS to determine the location

of data and the services, offered by the DSS, for a particular type of data or distributed data that has been processed

CSMS – The following subsystems are the CSMS subsystems, which interact with and support the SDPS to complete the ECS architecture.

2. The MSS with:

- hardware and software baseline and configuration management
- trouble ticketing and nonconformance report (NCR) tracking
- system start-up and shut-down
- fault and performance monitoring for networks, platforms, and software applications.
- user account management and user order tracking

3. The CSS with:

- Distributed Computing Environment (DCE) using TCP/IP for the network communications stack
- Libraries with common software mechanisms for application error handling, aspects of recovering client/server communications; Universal References to distributed objects and interfaces to e-mail, file transfer and network file copy capabilities.
- External gateways to translate from the ECS internal protocols, based on DCE, to protocols acceptable by external systems such as ASTER GDS and Landsat 7 LPS and vice versa.

4. The ISS with:

- Networking hardware devices (e.g., routers, switches, hubs, cabling, etc.) and their respective embedded software. For more information on site unique configurations, refer to the 920-series of General documents.

### **3.2.2.2 Multi-Site Architecture**

ECS Release 4 is distributed to five sites specified in Section 3.1.1. Each site is identified and its operational capabilities follow.

- EDC DAAC capabilities include:
  - ingest of ASTER Level 1A / 1B and Landsat 7 Level 0R data, with ancillary data needed for production
  - production, archival and distribution of ASTER products. EDC also archives and distributes the Landsat 7 0R data.

- GSFC DAAC capabilities include:
  - ingest of MODIS Level 0 data and related ancillary data
  - production, archival and distribution of the level 1A and 1B including the distribution of the level 1A, 1B and ancillary data to MODAPS for higher level processing..
- LaRC DAAC capabilities include:
  - ingest of CERES, MISR and MOPITT Level 0 and related ancillary data
  - production, archival, and distribution of the higher level products for MISR
  - data forwarding support for CERES through LaTIS and MOPITT to the SCF
- SMC capabilities include:
  - Overall ECS system performance monitoring, coordinating, and setting system-wide policies and priorities.

## 4. Subsystem Description

---

### Design Description Organization and Approach

This section presents a subsystem-by-subsystem overview description of the “as-built” ECS. The current detailed design information is provided for the Hardware Configuration Items (HWCI), Computer Software Component Items (CSCI), and Computer Software Components (CSC) for each subsystem and is being delivered to the DAACs in drop increments.

The ECS subsystem descriptions include:

- Subsystem functional overviews with a subsystem context diagram and a table of interface event descriptions
- CSCI descriptions with a context diagram and a table with interface event descriptions
- Architecture Diagrams, Process Descriptions, and Process Interface Event Tables. The Architecture Diagrams show the processes of the CSCI/CSC and how these processes connect with other CSCIs and CSCs of the same subsystem and the interfaces with other subsystems and external entities such as Operations, External Data Providers and Users. These processes and the supporting libraries are listed in Appendices A (Software Processes) and B (Software Libraries).
- Data Store descriptions for each CSCI in each ECS subsystem. The Data Stores are identified with the software name and shown in the architecture diagrams either as single data stores or as a group of data stores with a generic name such as “Data Stores” or “database.”
- Hardware descriptions of the subsystem hardware items and the fail-over strategy.

The convention used for Context and Architecture diagrams includes using circular and elliptical shapes to show the subject subsystems, CSCIs, CSCs, or processes (with name in bold) and squares and rectangles to show other subsystems, CSCIs, CSCs, and processes. Data stores are shown using the data store or database name with horizontal lines, one above and one below the name. An interface event is data, a message (which includes a notification or status), command/request or status code passed between subsystems, CSCIs, CSCs, or processes. The convention used to identify events is a straight line between two objects labeled with a phrase beginning with an action-oriented word to best describe the event. The arrow on the event line indicates an origination point and to where the event is directed. The response to this action is not shown in the diagram, in most cases, for simplification. However, the response is described in the description column of the interface event tables. These conventions are consistent with other ECS documentation. The convention for naming the ECS processes is *Ec* <subsystem abbreviation> meaningful name. The *Ec* identifies the process as an ECS developed process versus a COTS product. The *subsystem abbreviations* are listed subsystem-by-subsystem.

- Ds for DSS
- Dm for DMS
- Io for IOS
- Ms for MSS
- In for INS
- Pl for PLS
- Dp for DPS
- Cl for CLS
- Cs for CSS

The *meaningful name* identifies the process and its functionality within the subsystem, CSCI, or CSC. An example is `EcIoAdServer`, which identifies an ECS-developed IOS process called the Advertisement Server. Any names within an architecture diagram that do not follow this convention are COTS products. COTS product names are kept for simplicity and to adhere to licensing and trademark agreements.

## 4.1 Data Server Subsystem Overview

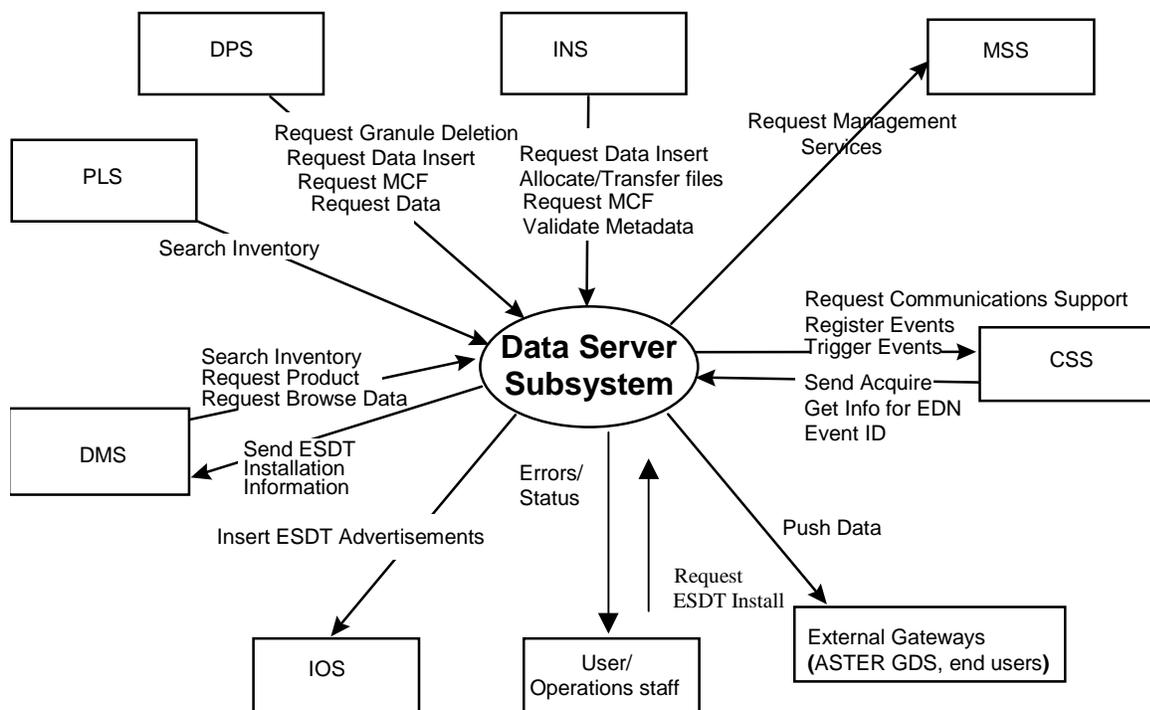
The Data Server Subsystem (DSS) provides capabilities to store, search, retrieve, and distribute earth science and related data. The DSS provides data repositories and management capabilities to safely store data on a permanent basis. The DSS stages data needed for data processing by the Data Processing Subsystem (DPS) or for retrieval by users at external locations. The DSS organizes and stores its data by data types, and provides advanced search capabilities and processing services on those data types in support of earth science data users. The DSS distributes data to users either electronically or on physical media. It also includes administrative capabilities to operate and manage its hardware and software.

DSS functionality includes:

- The DSS provides advertisements, for data types and corresponding data type services, to the Interoperability Subsystem (IOS)
- The DSS stores (archives) the Ingest Subsystem (INS) ingested data and products created via the DPS, and system management log file data
- The DSS receives service requests for data and data type services from external service requesters including the Client Subsystem (CLS), the Data Management Subsystem (DMS), the Planning Subsystem (PLS), and the DPS
- The DSS provides (distributes) data in response to service requests, to the request originator, by means of either electronic transfer or physical media. Alternatively, the subsystem can provide references to data as a Universal Reference (UR)

### Data Server Subsystem Context

Figure 4.1-1 is the context diagram for the DSS. The diagram shows the events DSS sends to other ECS subsystems and the events sent to DSS.



**Figure 4.1-1. Data Server Subsystem Context Diagram**

Table 4.1-1 provides a description of the interface events shown in the Data Server Subsystem context diagram.

**Table 4.1-1. Data Server Subsystem Interface Events (1 of 3)**

Event	Interface Event Description
Request Communications Support	The CSS provides a library of services available to each ECS subsystem. The services required to perform the specific subsystem assignments are requested by the subsystem from the CSS. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Register Events	The DSS sends the subscription events for an Earth Science Data Type (ESDT) to the CSS Subscription Server when an ESDT is installed into the system.
Trigger Events	The DSS notifies the CSS (via an event trigger) when a subscription event occurs on an ESDT Service.

**Table 4.1-1. Data Server Subsystem Interface Events (2 of 3)**

Event	Interface Event Description
Send Acquire	An “acquire” (instruction to obtain data) is created by the CSS and sent to the DSS. This is similar to the “Request Product” interface event except that it applies to EDOS expedited data.
Get Info for EDN	Expedited Data Set Notification (EDN) information is obtained from the DSS, by request, and used by the CSS.
Event ID	The CSS sends Event IDs to the DSS when ESDTs are installed.
Request Management Services	The MSS provides a basic management library of services to the subsystems, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The MSS forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>
Request Data Insert	The INS and DPS send requests to the DSS to insert a particular file or files into the archive. Inserted data is accompanied by metadata. The metadata is catalogued in the SDSRV inventory as a granule of a particular ESDT short name and version.
Allocate/Transfer files	The INS accesses and allocates and de-allocates disk space and transfers files to the DSS. A de-allocate request deletes data not marked as persistent data.
Request MCF	The INS and DPS can request the Metadata Configuration File (MCF) template, from the DSS, prior to a data insert request.
Validate Metadata	The DSS validates the metadata files that the INS has populated.
Request Granule Deletion	The DPS sends delete requests to the DSS for particular granules (interim data) in the metadata (the DSS inventory).
Request Product	The DMS forwards product requests (to the DSS) to acquire data products for an external user.
Request Data	The DPS sends requests to the DSS for a particular data granule retrieval. The product is transferred (pushed), via the File Transfer Protocol (FTP) service, onto the DPS science processor and used as input for Product Generation Executive (PGE) processing or for Science Software Integration and Test (SSIT) work.
Request ESDT Install	ESDT installation information for adding descriptor, dynamic link library (dll), version id, and archive configuration information for a new Earth Science Data Type.
Search Inventory	The PLS and DMS send Inventory Search Requests to the DSS to search the ECS Inventory (metadata). Earth Science Data Type (ESDT) universal references (URs) for the granules are returned in response from the DSS.
Request Browse Data	The DMS submits browse requests to the DSS to acquire reduced resolution products to support a product request.
Send ESDT Installation Information	The DSS sends ESDT installation information, to the DMS Data Dictionary, whenever a new ESDT is installed. This data consists of Valid and Collection level metadata.
Insert ESDT Advertisements	The Interoperability Subsystem (IOS) receives requests to insert advertisements for data types (ESDTs) from the DSS including both data product and signature service advertisements.

**Table 4.1-1. Data Server Subsystem Interface Events (3 of 3)**

Errors/Status	Error conditions and status of data distribution are sent from the DSS to the Operations staff or users.
Push Data	The DSS assembles instructions to send data and the DSS pushes data, via the FTP service, followed by a signal file to the destination specified in an acquire instruction by particular ESDTs that function this way.

### **Data Server Subsystem Structure**

The DSS contains three CSCIs:

- The Science Data Server (SDSRV) CSCI manages and provides user access to collections of non-document (non hard copy) earth science data, extracts and modifies data by request, accepts browse, search, and retrieval requests from users, and catalogs data insert requests from other ECS CSCIs, CSCs, and processes. The SDSRV CSCI manages earth science data as logical collections of related data, via interfaces independent of data formats and hardware configurations inherent in underlying storage technologies. The SDSRV manages interactive sessions with service requesters and informs the service requester of the availability of data and services via the IOS.
- The Data Distribution (DDIST) CSCI formats and distributes data to users electronically or via physical media. The formatting process includes the layout of a sequence of files, assembling a packing list and using Tape Archive (TAR) formats for tapes. The DDIST CSCI accepts requests from the SDSRV CSCI. The DDIST CSCI distributes staged data (data prepared for distribution), directs the Storage Management (STMGT) CSCI to transfer data either electronically or by 8mm tape, and provides a Graphical User Interface (GUI) for operations management to view, cancel, suspend/resume and change the priorities of distribution requests. Electronic distribution can be requested via an FTP push or pull. With push, the DDIST CSCI uses network resources managed by the STMGT CSCI to transfer the data to a remote destination specified by the requester. For pull, the data is placed in an area managed by the STMGT CSCI, from which the requester can retrieve the data.
- The Storage Management (STMGT) CSCI stores and manages data, and retrieves data files from the archives for other science data processing software. The STMGT CSCI provides an interface to make implemented changes in new data storage technologies transparent to users and without interfering with ECS systems outside the STMGT CSCI. The STMGT CSCI performs quality assurance processing and files recovery services. The STMGT CSCI also provides management of storage resources and prepares data for distribution.

The Data Server Subsystem hardware consists of the following four Hardware Configuration Items (HWCI):

- Access Control and Management

The Access Control and Management HWCI (ACMHW) is hardware to support the Ingest and Data Server Subsystems' software to directly interact with users. The ACMHW provides a level of security by isolating other hardware items from external software access.

- Working Storage

The Working Storage HWCI (WKSHW) is hardware to provide high-performance storage of large volumes of data on a temporary basis.

- Data Repository

The Data Repository HWCI (DRPHW) is hardware to provide high-capacity storage for long-term storage of data files.

- Distribution and Ingest Peripherals

The Distribution and Ingest Peripherals HWCI (DIPHW) is hardware to provide support to ingest and distribution via physical media.

Detailed information on hardware/software mapping, hardware diagrams, disk partitioning, etc., can be found in 920-TDx-00x, the 921-TDx-00x, and the 922-TDx-00x series of baseline documents.

### **Use of COTS in the Data Server Subsystem**

- RogueWave's Tools.h++

The Tools.h++ class libraries provide libraries of object strings and collections. These libraries must be installed for the DSS processes to run.

- RogueWave's DBTools.h++

The DBTools.h++ C++ class libraries interact with the Sybase database Structured Query Language (SQL) server and buffer the processes from the relational database used. These libraries must be installed for the Server to run.

- Rogue Wave's Net.h++

This is a C++ class library to provide an object-oriented interface to Inter-Process Communication (IPC) and network communication services. The Net.h++ framework enables developed code to be portable to multiple operating systems and network services. These libraries must be installed with the STMGT software to support interaction with other subsystems.

- Integrated Computer Solutions (ICS) Builder Xcessory

The Builder Xcessory GUI builder tool modifies the displays. The Builder Xcessory generates the C++ code to produce the Mtool display at run time. There is no operational part of the Builder Xcessory needed at run-time.

- Sybase SQL Server

Sybase's SQL server provides the capabilities to insert, update and delete database contents. The Sybase SQL Server must be operational to execute search and insert requests for metadata. The terms Sybase Server and Sybase SQL Server are used interchangeably in this section.

- Autometric's Spatial Query Server

The Spatial Query Server (SQS) provides the capability to store and search spatial metadata. SQS has spatial indexing to search on spatial metadata for the SDSRV.

- Sybase Open Client / CT\_LIB

The Sybase Open Client provides access between DSS custom code and the Sybase SQL Server DBMS.

- University of Illinois' Hierarchical Data Format (HDF)

HDF provides EOS extended capabilities for sub-setting services with the SDSRV CSCI.

- University of Colorado's Object Description Language (ODL)

ODL provides a general architecture, independent means of passing metadata files between subsystems.

- DCE Client

DCE Client provides DSS with communications between other subsystems. DCE can reside on one or both sides of the interface. An instance must be installed on the platform where DSS resides. Although the DCE Client is part of CSS, this COTS item must be installed for DSS to run in the ECS operational and test environment.

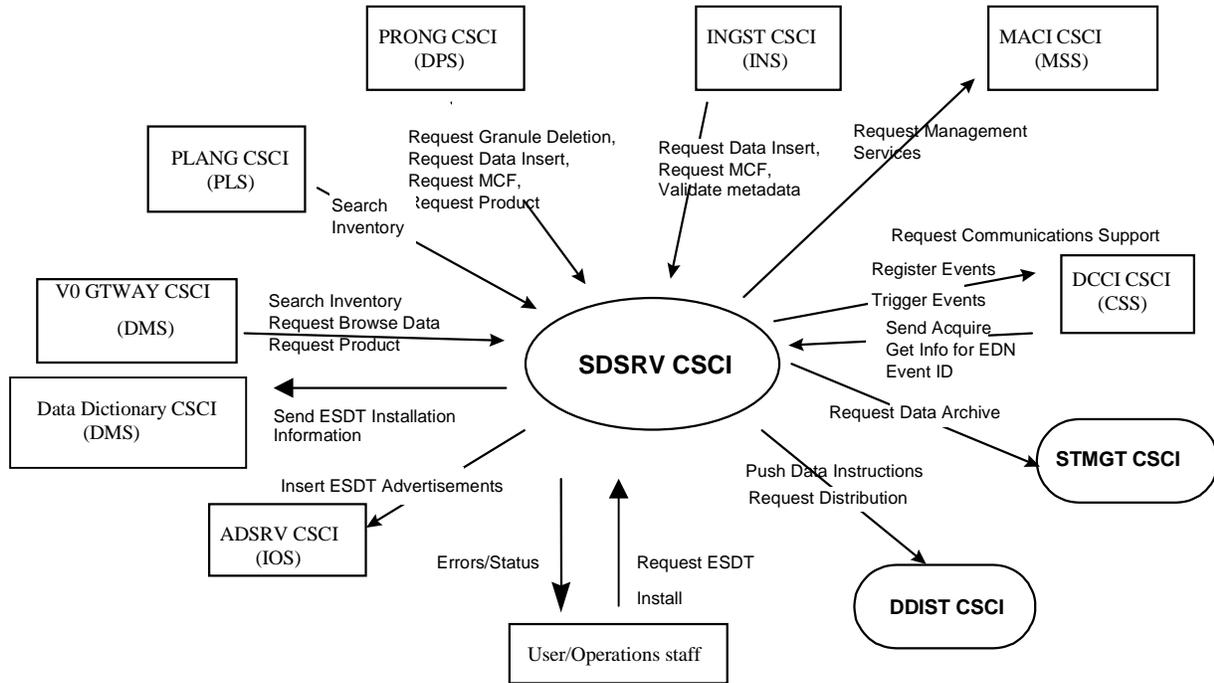
#### **4.1.1 Science Data Server Software Description**

##### **4.1.1.1 Science Data Server Functional Overview**

The SDSRV CSCI provides the ECS with a catalog of Earth Science Data holdings, and the Earth Science Data Type services that operate on the data. The SDSRV CSCI provides a catalog of metadata describing the archived data holdings of the ECS and provides mechanisms to acquire the data from the archive. The SDSRV CSCI also provides data type services on the catalog and a data reduction or sub-setting service.

#### 4.1.1.2 Science Data Server Context

Figure 4.1.1.2-1 is the SDSRV CSCI context diagram. The diagram shows the events sent to the SDSRV CSCI and the events the SDSRV CSCI sends to other CSCIs. The events have been grouped by CSCI including the Storage Management and Data Distribution functions of the DSS.



**Figure 4.1.1.2-1. SDSRV CSCI Context Diagram**

Table 4.1.1.2-1 provides descriptions of the interface events shown in the SDSRV CSCI context diagram.

**Table 4.1.1.2-1. SDSRV CSCI Interface Events (1 of 3)**

Event	Interface Event Description
Register Events	The SDSRV CSCI sends the subscription events for an Earth Science Data Type to the CSS Subscription Server when an ESDT is installed into the system.
Trigger Events	The SDSRV CSCI sends event triggers when a subscription event has occurred on an Earth Science Data Type Service.
Send Acquire	An "acquire" (instruction to obtain data) is created by the CSS (E-mail Parser Gateway CSC or the Subscription Server CSC) and sent to the DSS (SDSRV CSCI).

**Table 4.1.1.2-1. SDSRV CSCI Interface Events (2 of 3)**

Event	Interface Event Description
Get Info for EDN	Information for the Expedited Data Set Notification (EDN) is obtained from the DSS (SDSRV CSCI), by request, and used by the CSS (E-mail Parser Gateway CSC).
Event ID	The CSS Subscription Server (SBSRV CSC) sends Event IDs to the SDSRV CSCI when ESDTs are installed.
Request Granule Deletion	The PRONG CSCI sends delete requests to the SDSRV CSCI for particular granules (interim data) in the SDSRVmetadata.
Request Data Insert	The INGST AND PRONG CSCIs send requests to the SDSRV CSCI to insert a particular file or files into the archive. Inserted data is accompanied by metadata. The metadata is catalogued in the SDSRV inventory as a granule of a particular ESDT short name and version. For INS, this data can be algorithms, Level 0 (L0) data, standard products, ancillary data, correlative data or calibration data. For the DPS, these files can be processing output, static files received with PGEs, PGE Tape Archive (TAR) files, Algorithm Packages (APs), Science Software Archive Packages (SSAPs), or Delivered Algorithm Packages (DAPs), failed PGE tar files, or production history files.
Request MCF	The INGST AND PRONG CSCIs request the Metadata Configuration File (MCF) template, from the DSS, prior to a data insert request. Also, the PRONG CSCI can request from the SDSRV CSCI the MCF for a particular ESDT short name prior to a data insert request.
Request Data	The PRONG CSCI submits requests to the SDSRV CSCI for a particular data granule to be provided. The data is pushed, via the FTP service, onto the DPS science processor to be used as input for PGE processing or for SSIT work.
Search Inventory	The PLANG AND V0 GTWAY CSCIs send Inventory Search Requests to the SDSRV CSCI to search the ECS Inventory (archives). ESDT universal references (URs) for the granules are returned in response from the SDSRV CSCI.
Request Browse Data	The V0 GTWAY CSCI submits browse requests to the SDSRV CSCI to acquire reduced resolution products to support a product request.
Send ESDT Installation Information	The SDSRV CSCI sends ESDT installation information, to the DDICT CSCI, whenever a new ESDT is installed. This data consists of Valids and Collection level metadata.
Validate metadata	The SDSRV CSCI validates the metadata files, which the INGST CSCI has populated.
Insert ESDT Advertisements	The ADSRV CSCI receives requests to insert advertisements for data types (ESDTs) from the SDSRV CSCI including both data product and signature service advertisements.
Request Distribution	The SDSRV CSCI sends distribution requests to the DDIST CSCI for various categories of data. The distribution services on those data are essentially identical for all data categories.
Request Data Archive	The SDSRV CSCI sends archive requests to the STMGT CSCI for various categories of data. The archive services on those data are essentially identical for all data categories and consist of requests to retrieve files, store files, copy files, or create/destroy a staging disk.

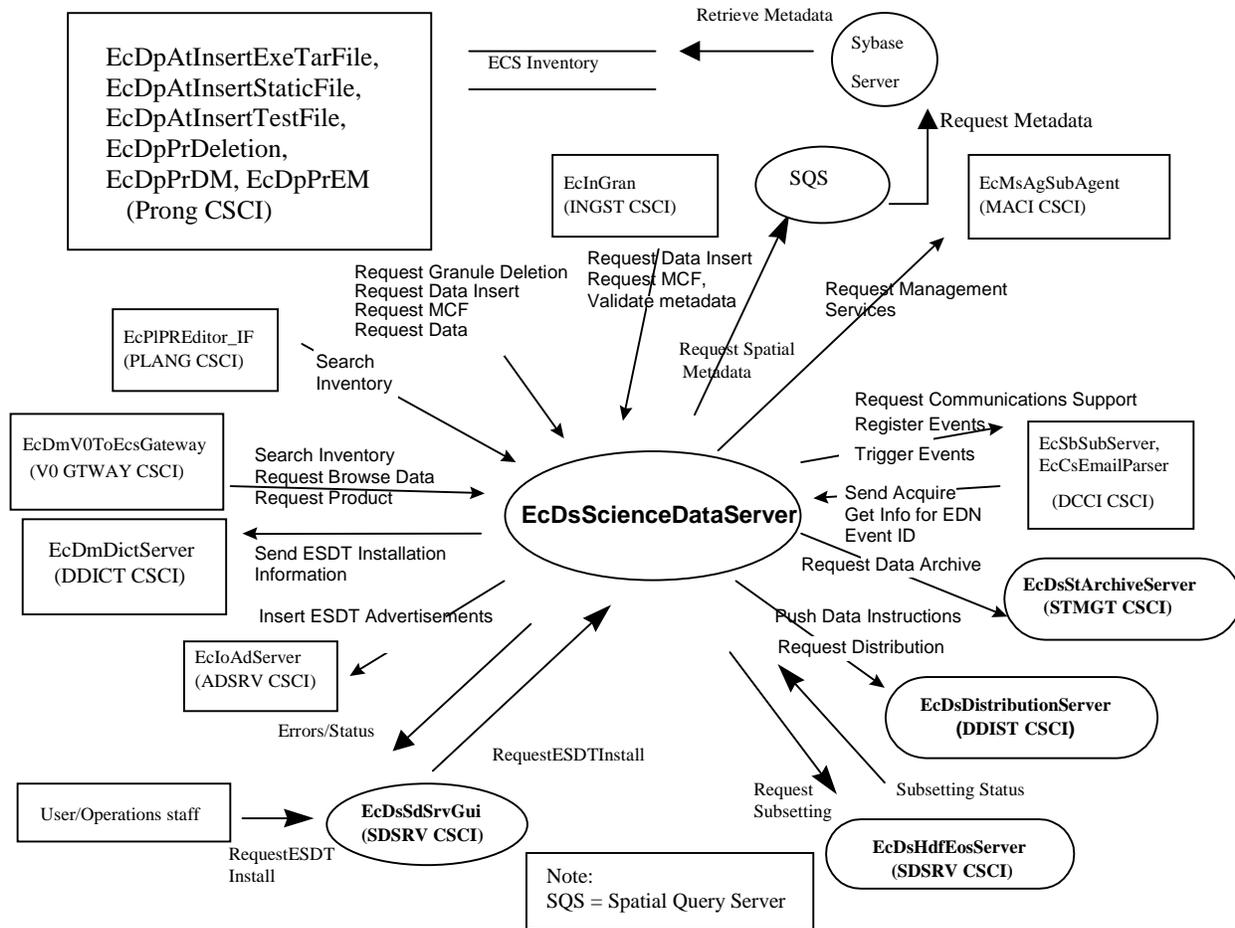
**Table 4.1.1.2-1. SDSRV CSCI Interface Events (3 of 3)**

Event	Interface Event Description
Errors/Status	Error conditions and status of data distribution are sent from the SDSRV CSCI (for acquires that are synchronous) or the DDIST CSCI (if the acquire is asynchronous and the DDIST CSCI gets the request) to the Operations staff or users.
Request ESDT Install	ESDT installation information for adding the descriptor, dynamic link library (dll), version id, and archive configuration information for a new Earth Science Data Type.
Request Communications Support	The DCCI CSCI provides a library of services available to each ECS CSCI. The services required to perform the specific CSCI assignments are requested by the CSCI from the DCCI CSCI. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Request Management Services	The MACI CSCI provides a basic management library of services to the CSCIs, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> – The MACI CSCI forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>
Push Data Instructions	The SDSRV CSCI assembles instructions to send data and sends the instructions to the DDIST CSCI. The DDIST CSCI sends a request to the STMGT CSCI to push the data, via the FTP service, followed by a signal file (Note: very few acquires involve a signal file), to the destination specified in an acquire instruction.
Request Product	The V0 Gateway CSCI forwards requests to the SDSRV CSCI for a product order from an external user to be distributed by the DDIST CSCI upon receipt of the data from the STMGT CSCI.

#### 4.1.1.3 Science Data Server Architecture

Figure 4.1.1.3-1 is the SDSRV CSCI Architecture diagram. The diagram shows the events sent to the SDSRV CSCI processes and the events the SDSRV CSCI processes send to other processes.

The Science Data Server (SDSRV) CSCI is five processes: three ECS custom developed processes and two COTS processes. The three ECS custom developed processes are the Science Data Server (EcDsScienceDataServer), the Hierarchical Data Format (HDF) EOS Server (EcDsHdfEosServer) [Note: multiple HDF Server processes can be defined.], and the Science Data Server GUI (EcDsSdSrvGui). The COTS processes are the Sybase SQL Server and the Spatial Query Server (SQS). The SDSRV CSCI uses the Sybase SQL Server Database Management System (DBMS) for ECS Inventory and Configuration data storage. The server holds Earth Science Data Type configuration information and the data catalog for all the archived products found at a DAAC.



**Figure 4.1.1.3-1. SDSRV CSCI Architecture Diagram**

#### 4.1.1.4 Science Data Server Process Descriptions

Table 4.1.1.4-1 provides descriptions of the processes shown in the SDSRV CSCI architecture diagram.

**Table 4.1.1.4-1. SDSRV CSCI Processes (1 of 2)**

Process	Type	COTS / Developed	Functionality
EcDsScienceDataServer	Server	Developed	<p>The EcDsScienceDataServer server manages collections of earth science and related data, and service requests for the storage, search, retrieval, and manipulation of data within those collections. The science data server performs the following functions:</p> <ul style="list-style-type: none"> <li>• manages earth science data as logical collections of related data, using interfaces independent of any data formats and hardware configurations provided by underlying storage technologies,</li> <li>• manages interactive sessions with users,</li> <li>• manages the processing of service requests from the DMS (VO Gateway), providing a variety of services on earth science and related data,</li> <li>• issues requests to the STMG and DDIST CSCIs to perform storage and distribution services in support of processing service requests,</li> <li>• provides advertisements to the IOS to announce the availability of data and services to users,</li> <li>• manages the processing of service requests from the INS and DPS to "insert" data for long-term storage and access,</li> <li>• manages the processing of service requests from the DPS to provide data to be used as input for data processing,</li> <li>• provides subscription events and event triggers to the CSS subscription server</li> <li>• provides sub-setting requests to the EcDsHdfEosServer for sub-setting science data files.</li> </ul> <p>The EcDsScienceDataServer supports:</p> <ul style="list-style-type: none"> <li>• Single requests, one at a time</li> <li>• Multiple concurrent requests</li> <li>• Asynchronous request processing</li> <li>• Request processing buffered from Remote Procedure Call (RPC) threads</li> <li>• Multiple threads within a single user session</li> </ul>
EcDsHdfEosServer	Server	Developed	<p>The EcDsHdfEosServer provides science data sub-setting capabilities for earth science data configured with a sub-setting service.</p> <p>EcDsHdfEosServer supports:</p> <ul style="list-style-type: none"> <li>• Single requests, one at a time</li> <li>• Asynchronous request processing</li> <li>• Request processing buffered from RPC threads</li> </ul>

**Table 4.1.1.4-1. SDSRV CSCI Processes (2 of 2)**

Process	Type	COTS / Developed	Functionality
EcDsHdfEosServer	Server	Developed	The EcDsHdfEosServer provides science data sub-setting capabilities for earth science data configured with a sub-setting service. EcDsHdfEosServer supports: <ul style="list-style-type: none"> <li>• Single requests, one at a time</li> <li>• Asynchronous request processing</li> <li>• Request processing buffered from RPC threads</li> </ul>
EcDsSdSrvGui	GUI	Developed	The EcDsSdSrvGui provides an operator interface for : <ul style="list-style-type: none"> <li>• receiving descriptor files and dynamic link libraries (dll) for configuring ESDTs into the EcDsScienceDataServer</li> <li>• monitoring active EcDsScienceDataServer requests</li> </ul> The EcDsSdSrvGui supports: <ul style="list-style-type: none"> <li>• Single requests, one at a time</li> </ul>
SYBASE	Server	COTS	Provides the management of spatial data types of an earth science catalog of metadata for the ECS. Includes capabilities for searching and storing the catalog.
Spatial Query Server	Server	COTS	Provides the capability to manage spatial data types of earth science catalog metadata for the ECS (including specialized spatial searches).

#### 4.1.1.5 Science Data Server Process Interface Descriptions

Table 4.1.1.5-1 provides descriptions of the interface events shown in the SDSRV CSCI architecture diagram.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (1 of 9)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request Distribution	One per distribution request	<i>Library:</i> DsDdC <i>Class:</i> DsDdRequestMgrC	<i>Process:</i> EcDsScienceDataServer <i>Library:</i> DsSr <i>Class:</i> DsSrWorkingCollection	The EcDsScienceDataServer sends distribution requests to the EcDsDistributionServer for various categories of data. The distribution services are essentially identical for all data categories.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (2 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Register Events	One per ESDT installation	<i>Library:</i> EcSbSrSh <i>Class:</i> EcSbEvent	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDeEventCustomizer	The EcDsScienceDataServer sends the subscription events for an Earth Science Data Type to the EcSbSubServer when an ESDT is installed into the system.
Trigger Events	One per subscription event	<i>Library:</i> EcSbCI <i>Class:</i> EcCIEvent	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsBtSbsrvNotifier	The EcDsScienceDataServer sends event triggers when a subscription event has occurred on an Earth Science Data Type Service.
Send Acquire	One per acquire	<i>Library:</i> DsCI <i>Classes:</i> DsCIRequest DsCICommand DsCIESDTReferenceCollector	<i>Library:</i> EcSbSv <i>Class:</i> EcSbSubscription	An "acquire" (instruction to obtain data) is created by the EcCsEmailParser or the EcSbSubServer and sent to the EcDsScienceDataServer.
Get Info for EDN	One per request	<i>Library:</i> DsCI <i>Class:</i> DsCIESDTReference	<i>Process:</i> EcCsEmailParser	Information for the Expedited Data Set Notification (EDN) is obtained from the EcDsScienceDataServer and used by theEcCsEmailParser.
Event ID	One per ESDT installation	<i>Library:</i> EcSbSrSh <i>Class:</i> EcSbEvent	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDeEventCustomizer	The EcSbSubServer sends Event IDs to the EcDsScienceDataServer when ESDTs are installed.
Request Data Archive	One per Archive request	<i>Libraries:</i> DsStCmn DsStSt <i>Classes:</i> DsStStagingDisk DsStStagingDiskController	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsGeESDT	The EcDsScienceDataServer sends Archive requests to the EcDsStArchiveServer for various categories of data. The archive services are essentially identical for all data categories, and consist of requests to Retrieve files, Store files, Copy files, or Create/Destroy Staging Disk.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (3 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Request ESDT Install	One per new ESDT installation	<i>Library:</i> DsAd <i>Class:</i> DsAdDataTypeCollector	<i>Process:</i> EcDsSdSrvGui <i>Class:</i> DsGuSdDataType	ESDT installation information for adding the descriptor, dynamic link library (dll), version ID, and archive configuration information for a new ESDT.
Request Metadata	One per request to store, search, delete, update Earth Science Metadata	<i>Library:</i> DsDb <i>Class:</i> DsDbInterface via the Sybase Open Server, SQS and Sybase Open Client COTS.	<i>Process:</i> Spatial Query Server (SQS)	Requests to store, search, delete, or update Earth Science metadata in the SDSRV database. Metadata Results are sent back to the Spatial Query Server. The results include the status of SQL Server commands.
Retrieve Metadata	One to many per metadata request	<i>Libraries (Sybase):</i> Libtcl.so Libtli.so Libsybdb.so <i>Class:</i> Sybase Open Client /ct_lib	<i>Process:</i> Sybase SQL Server	The Sybase SQL Server retrieves metadata from the SDSRV Inventory database.
Request Subsetting	One per request to reduce resolution	<i>Library:</i> DsCsSh <i>Classes:</i> DsCsConformant DsCsNonConformant	<i>Process:</i> EcDsScienceDataServer <i>Classes:</i> DsESDTLsL70R DsESDTLsL70RWRS	Requests to reduce the resolution of an archived earth science data product using configured services for spatial / temporal reduction.
Subsetting status	One per sub-setting request	<i>Library:</i> DsCsSh <i>Classes:</i> DsCsConformant DsCsNonConformant	<i>Process:</i> EcDsScienceDataServer <i>Classes:</i> DsESDTLsL70R DsESDTLsL70RWRS	Status of sub-setting requests.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (4 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Request Spatial Metadata	One per request to store, search, delete, or update spatial Earth Science Metadata.	<i>Library:</i> DsDb <i>Class:</i> DsDbInterface	<i>Process:</i> EcDsScienceDataServer <i>Library:</i> DpPrDssIF <i>Class:</i> DsDbInterface	Requests to store, search, delete, or update spatial Earth Science metadata in the SDSRV database. Metadata Results are sent back to the EcDsScienceDataServer. The results include the status of the SQL Server and SQS Server commands. The SQS handles the translation of spatial metadata data types (understood by the EcDsScienceDataServer and SQS) to relational data types (understood by SQS and the Sybase SQL Server) and vice versa.
Request Granule Deletion	One per granule delete request	<i>Library:</i> DsCI <i>Class:</i> DsCIRequest DsCICommand DsCIESDTReferenceCollector	<i>Process:</i> EcDpPrDeletion <i>Library:</i> DpPrDssIF <i>Class:</i> DpPrDSSInterface	The EcDpPrDeletion sends delete requests to the EcDsScienceDataServer for particular granules (interim data) in the SDSRV archives.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (5 of 9)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request Data Insert	One per data insert request from EcDpPrDM or EclnGran	<i>Library:</i> DsCI <i>Class:</i> DsCIRequest, DsCICommand	<i>Processes:</i> EcDpPrDM, EcDpPrEM, EcDpAtInsertExeTarFile, EcDpAtInsertStaticFile, EcDpAtInsertTestFile  <i>Process:</i> EclnGran <i>Library:</i> InPreprocess <i>Class:</i> InDataServerInsertionTask	The EclnGran, EcDpPrEM, EcDpAtInsertExeTarFile, EcDpAtInsertStaticFile, EcDpAtInsertTestFile, and EcDpPrDM processes send requests to the EcDsScienceDataSserver to insert a particular file or files into the archive. Inserted data is accompanied by metadata. The metadata is catalogued in the SDSRV inventory as a granule of a particular ESDT short name and version. For EclnGran, this data can be algorithms, Level 0 (L0) data, standard products, ancillary data, correlative data or calibration data. For the EcDpPrDM, these files can be processing output, static files received with PGEs, PGE Tape Archive (TAR) files, Algorithm Packages (Aps), Science Software Archive Packages (SSAPs), or Delivered Algorithm Packages (DAPs), failed PGE tar files, or production history files.
Request Product	One per product order.	<i>Library:</i> DsCI <i>Classes:</i> DsCIESDTReferenceCollector, DsCIRequest, DsCICommand	<i>Process:</i> EcDmV0ToEcsGateway <i>Library:</i> RequestProcessing <i>Class:</i> DmGwAcquireRequest	The EcDmV0ToEcsGateway forwards requests from an external user to be distributed by the EcDsDistributionServer upon receipt of the data from the EcDsStStagingDiskServer.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (6 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Request MCF	One per set of external data received by ECS	<i>Library:</i> DsCI <i>Class:</i> DsCIDescriptor	<i>Processes:</i> EcDpPrEM, EcDpAtGetMCF <i>Library:</i> DpPrDsslF <i>Class:</i> DpPrDSSInterface  <i>Process:</i> EcInGran <i>Library:</i> InPreprocess <i>Class:</i> InDataPreprocessTask	The EcInGran and EcDpPrEM processes request the Metadata Configuration File (MCF) template, from the EcDsScienceDataServer, prior to a data insert request.
Request Data	One per data granule request from DPS or DMS	<i>Library:</i> DsCI <i>Classes:</i> DsCIESDTReferenceCollector, DsCIRequest, DsCICommand	<i>Process:</i> EcDpPrQaMonitorGUI <i>Library:</i> DpPrQaMonitor <i>Class:</i> DpPrQaMonitor  <i>Processes:</i> EcDpPrDM, EcDpPrEM, EcDpAtSSAPGui, EcDpAtStageDAP <i>Library:</i> DpPrDsslF <i>Class:</i> DpPrDSSInterface	The EcDpPrDM process submits requests to the EcDsScienceDataServer for a particular data granule to be provided. The data is pushed, via the FTP service, onto the DPS science processor to be used as input for PGE processing or for SSIT work.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (7 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Search Inventory	One per service request	<i>Library:</i> DsCI <i>Classes:</i> DsCIESDTReferenceCollector, DsCIRequest, DsCICommand	<i>Process:</i> EcPIPREditor_IF <i>library:</i> PICore1 <i>class:</i> PIDataType <i>Process:</i> EcDmV0ToEcsGateway <i>Library:</i> RequestProcessing <i>Class:</i> DmGwInvSearchRequest	The EcPIPREditor_IF and EcDmV0ToEcsGateway processes send Inventory Search Requests to the EcDsScienceDataServer to search the ECS Inventory (archives). ESDT universal references (URs) for the granules are returned in response from the EcDsScienceDataServer.
Request Browse Data	One per browse request	<i>Library:</i> DsCI <i>Classes:</i> DsCIESDTReferenceCollector, DsCIRequest, DsCICommand	<i>Library:</i> RequestProcessing <i>Class:</i> DmGwBrowseRequest	The EcDmV0ToEcsGateway submits browse requests to the EcDsScienceDataServer to acquire reduced resolution products to support a product request.
Send ESDT Installation Information	One per new ESDT installation	<i>Library:</i> EcDmDdClient <i>Class:</i> DmDdCISchemaRequest	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDeDataDictController	The EcDsScienceDataServer sends ESDT installation information, to the EcDmDictServer, whenever a new ESDT is installed. This data consists of Valid and Collection level metadata.
Insert ESDT Advertisements	One per advertisement insert request	<i>Libraries:</i> IoAdCore IoAdSubs <i>Classes:</i> IoAdApprovedAdv IoAdGroup IoAdProvider IoAdProduct	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDeIOSController	The EcIoAdServer receives requests to insert advertisements for data types (ESDTs) from the EcDsScienceDataServer including both data product and signature service advertisements.

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (8 of 9)**

Event	Event Frequency	Interface	Initiated By	Event Description
Errors/Status	Once per distribution request completion	<i>Library:</i> EcUt <i>Class:</i> EcLgErrorMsg	<i>Process:</i> EcDsScienceDataServer	Error conditions and status of data distribution are sent from the EcDsScienceDataServer (for acquires that are synchronous) or the EcDsDistributionServer (if the acquire is asynchronous and the EcDsDistributionServer gets the request) to the Operations staff or users.
Request Management Services	One service per request	<i>Library:</i> EcAgInstrm <i>Classes:</i> EcAgManager, MsAgRegistry	<i>Process:</i> EcDsScienceDataServer	The EcMsAgSubAgent provides a basic management library of services to the CSCIs, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> – The EcMsAgSubAgent forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>

**Table 4.1.1.5-1. SDSRV CSCI Process Interface Events (9 of 9)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Push Data Instructions	One per distribution request	<p><i>Libraries:</i> DsDdB, DsDdC, DsDdl</p> <p><i>Classes:</i> DsDdRequestMgrC, DsDdGranuleC</p>	<p><i>Process:</i> EcDsScienceDataServer</p> <p><i>Library:</i> DsSr</p> <p><i>Class:</i> DsSrWorkingCollection</p>	The EcDsScienceDataServer assembles instructions to send data and sends the instructions to EcDsDistributionServer. The EcDsDistributionServer sends a request to the EcDsStStagingDiskServer to push the data, via the FTP service, followed by a signal file, to the destination specified in an acquire instruction.
Request Communications Support	One service per request	To simplify the interface table for the Request Communications Support event, refer to the CSS sections of Appendix B, Software Libraries, for the libraries and classes used as the interfaces to fulfill the services requested by ECS processes.	<p><i>Process:</i> EcDsScienceDataServer</p>	The CSS Process Framework provides a library of services available to each ECS process. The services required to perform the specific process assignments are requested by the process from the Process Framework. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Validate metadata	One per data insert request.	<p><i>Library:</i> DsCI</p> <p><i>Class:</i> DsCIDescriptor</p>	<p><i>Process:</i> EcInGran</p> <p><i>Library:</i> InPreprocess</p> <p><i>Class:</i> InDataPreprocessTask</p>	The EcDsScienceDataServer validates the metadata files that the EcInGran process has populated.

#### 4.1.1.6 Science Data Server Data Stores

Table 4.1.1.6-1 provides a description of the data stores for the SDSRV CSCI, and the conceptual model of the data store. The physical model for the SDSRV data stores can be found in the Science Data Server Database Design and Schema Specifications for the ECS Project (CDRL 311).

**Table 4.1.1.6-1. SDSRV CSCI Data Stores**

Data Store	Type	Description
ECS Inventory	Database	<p>The ECS Inventory (archives) contains the metadata describing the earth science data for the Earth Science Data Types at a specific DAAC. The metadata describes:</p> <ul style="list-style-type: none"> <li>• Collection level information</li> <li>• Browse data</li> <li>• Science data (as granules)</li> <li>• Quality Assessments</li> <li>• Algorithm Packages</li> <li>• Delivered Algorithm Packages</li> <li>• Production History</li> </ul> <p>The ECS catalog also contains systems data for the dynamic configuration of the EcDsScienceDataServer.</p> <p>The ECS catalog also contains implementation of the “ECS Data Model” for Attribute Validity checking.</p> <p>The ECS catalog also contains system data for ESDT Configuration.</p>

#### 4.1.2 Data Distribution Software Description

##### 4.1.2.1 Data Distribution Functional Overview

The Data Distribution (DDIST) CSCI monitors and controls processing for distribution requests. Data Distribution processing consists of directing the STMGT CSCI to place data for distribution in working storage and creating packing lists, and directing the STMGT CSCI to copy data on to tape or push data as required via the FTP, and sending notifications for pulls completed via the FTP. Data handled electronically is either pushed via the FTP to a user-specified location or placed in a directory to be pulled. If data is to be pulled, once the data is ready, the DDIST CSCI sends an electronic message to the user providing the required information for the user to pull the data. If data is to be distributed via tape, a complete packing list is generated as well as an inventory list for each tape generated. The DDIST CSCI has a GUI interface with the administration/operations staff (Admin./Ops). The GUI provides error conditions and status to operations staff and enables the operations staff to set parameters and control operations including suspending, canceling, and resuming requests, changing the priorities of requests, performing multiple selects, and setting threshold sizes. The DDIST CSCI provides limited automatic error response by suspending requests when most errors are encountered. The physical media type supported is 8mm tape.

The DDIST CSCI determines the number of tapes for an order, the files to go on the tape(s), generates packing lists, and tape inventory lists. The DDIST CSCI directs the STMGT CSCI to pack one tape for a given request at a time.

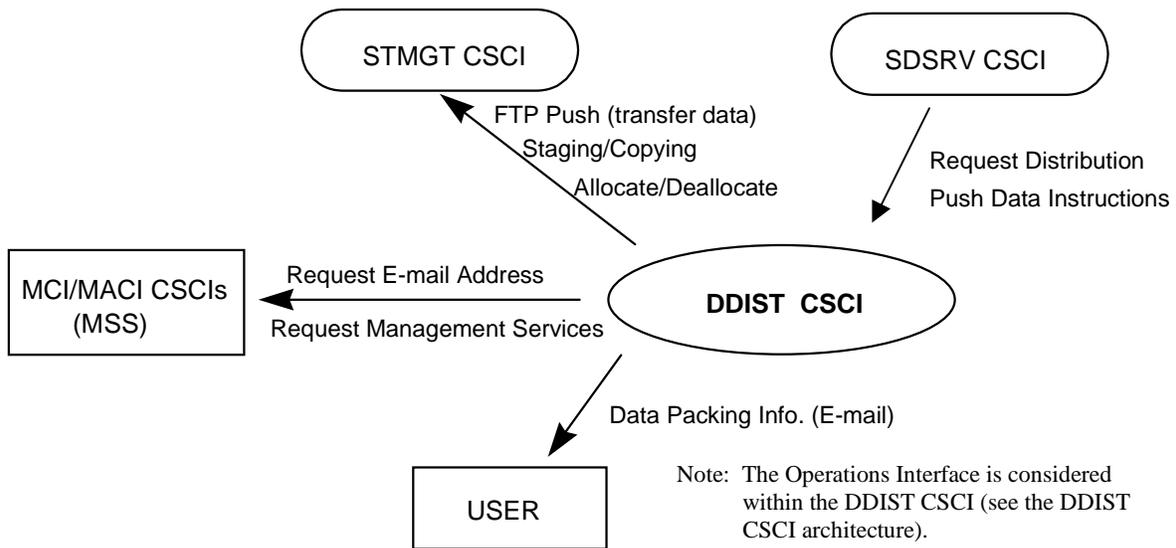
The DDIST CSCI has an interface with the following:

- STMGT CSCI
- SDSRV CSCI
- Administrator/Operations staff (through GUI or command line)
- MSS (MCI/MACI CSCIs)

The Administrator/Operations staff and the SDSRV CSCI control the DDIST CSCI activities. The DDIST CSCI receives direction to perform its functions from calls to the STMGT CSCI.

#### 4.1.2.2 Data Distribution Context

Figure 4.1.2.2-1 is the DDIST CSCI context diagram. The diagram shows the events sent to the DDIST CSCI and the events the DDIST CSCI sends to other CSCIs and Users. Table 4.1.2.2-1 provides descriptions of the events (by service name) shown in the DDIST CSCI context diagram.



**Figure 4.1.2.2-1. DDIST CSCI Context Diagram**

**Table 4.1.2.2-1. DDIST CSCI Interface Events (1 of 2)**

Event	Interface Event Description
Request E-mail Address	The DDIST CSCI requests E-mail addresses from the MCI CSCI. These addresses are used to send E-mail to the User and the SDSRV CSCI.

**Table 4.1.2.2-1. DDIST CSCI Interface Events (2 of 2)**

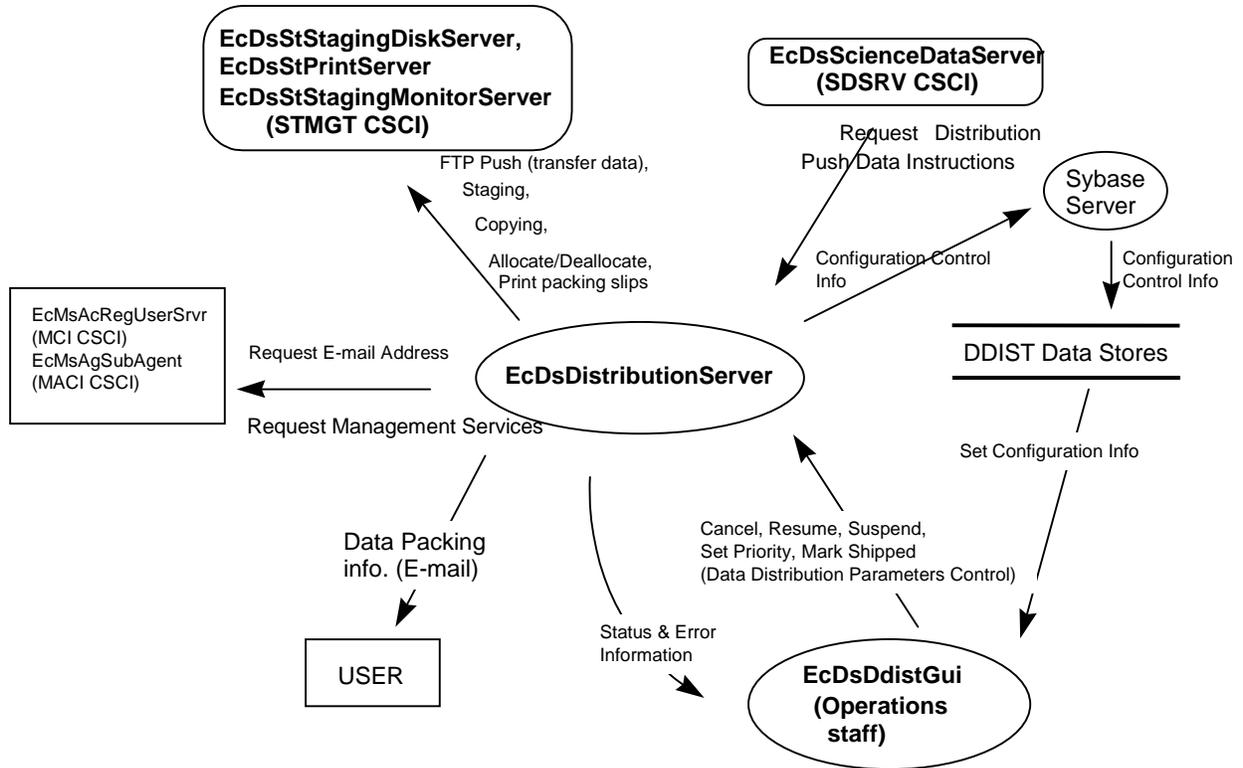
Event	Interface Event Description
FTP Push	The DDIST CSCI requests the push of granules staged by the STMGT CSCI via the FTP Service.
Request Management Services	<p>The MACI and MCI CSCIs provide a basic management library of services to the CSCIs, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include:</p> <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The MACI CSCI forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul> <p>The MCI CSCI also interfaces with other CSCIs or CSCs to perform the following:</p> <ul style="list-style-type: none"> <li>• <b>User Profile Request</b> – The MCI CSCI provides requesting CSCIs or CSCs with access to user profile information such as e-mail address and shipping address to support their processing activities.</li> </ul>
Copying	The DDIST CSCI requests copy of granules staged by the STMGT CSCI.
Staging	Data is staged on disk to be pushed or distributed to a user or to be stored in the data archives (inventory).
Allocate/De-allocate	Space allocations and de-allocations for data distributed or having been distributed, respectively.
Request Distribution	A request sent from the SDSRV CSCI to the DDIST CSCI for science data or a product to be sent to a specified user.
Data Packing Info. (E-mail)	Notification sent to the user via E-mail about a data product to be or has been distributed. Data handled electronically is either pushed, via the FTP service, to a user-specified location or placed in a directory to be pulled via the FTP service by the user.
Push Data Instructions	The SDSRV CSCI assembles instructions to send data and sends the instructions to the DDIST CSCI. The DDIST CSCI sends a request to the STMGT CSCI to push the data, via the FTP service, followed by a signal file, to the destination specified in an acquire instruction.

### 4.1.2.3 Data Distribution Architecture

The DDIST CSCI is two ECS developed processes with the addition of the Sybase Server COTS hardware and software process package as a data repository identified as:

- EcDsDistributionServer - Data Distribution
- EcDsDdistGui - Data Distribution GUI
- Sybase Server - Data Repository (storage area)

Figure 4.1.2.3-1 is the DDIST CSCI architecture diagram. The diagram shows the events sent to the DDIST CSCI processes and the events the DDIST CSCI processes send to other processes and the Operations staff.



**Figure 4.1.2.3-1. DDIST CSCI Architecture Diagram**

#### 4.1.2.4 Data Distribution Process Descriptions

Table 4.1.2.4-1 provides descriptions of the processes shown in the DDIST CSCI architecture diagram.

**Table 4.1.2.4-1. DDIST CSCI Processes**

Process	Type	COTS/ Developed	Functionality
EcDsDistribution Server	Server	DEV	This process provides the control and coordination for data distribution through request processing.
EcDsDdistGui	GUI	DEV	This process enables operations to initiate, track, and manipulate distribution requests by using input GUI controls and database information.
Sybase	Server	COTS	The process contains the request list and has a set of stored procedures, which updates the request configuration, provides the request configuration to GUI operations and checkpoints the state of the CSCI for fault recovery purposes.

#### 4.1.2.5 Data Distribution Process Interface Descriptions

Table 4.1.2.5-1 provides descriptions of the interface events shown in the DDIST CSCI architecture diagram.

**Table 4.1.2.5-1. DDIST CSCI Process Interface Events (1 of 4)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request E-mail Address	One per address request	<i>Library:</i> DsDdSSh <i>Class:</i> DsDdMedia	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia	E-mail addresses are requested through the EcDsDistributionServer from the EcMsAcRegUserSrvr. These addresses are used for e-mail to the user.
FTP Push	One per data store	<i>Library:</i> DsDdFtpPush <i>Class:</i> DsDdFtpPushMedia	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdScheduler	Data stored to a staging area prior to distribution.
Copying	One per data copy operation	<i>Library:</i> DsDdS <i>Class:</i> DsDdGranuleS	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules	Data copied from the data archives for distribution.
Staging	One per data stage operation	<i>Library:</i> DsDdS <i>Classes:</i> DsDdMediaDist DsDdGranuleS	<i>Process:</i> Server <i>Class:</i> DsDdGranules	Data is staged on disk to be pushed or distributed to a user or stored in the data archives.
Allocate/Deallocate	One per space allocation or de-allocation	<i>Library:</i> DsDdS <i>Class:</i> DsDdMedia	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia	Space allocations and de-allocations for data distributed or having been distributed, respectively.
Request Distribution	One per request for science data or product order	<i>Library:</i> DsDdSSh <i>Classes:</i> DsDdScheduler, DsDdRequestMgrReal, DsDdDCERequestMgr Concrete	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDdRequestMgrReal	A request sent from the EcDsScienceDataServer to the EcDsDistributionServer for science data or a product to be sent to a specified user.

**Table 4.1.2.5-1. DDIST CSCI Process Interface Events (2 of 4)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Configuration Control Info	One per configuration sent	<i>Library:</i> DsDdSSH <i>Classes:</i> DsDdDistListS, DsDdDistFileS, DsDdDistRequestS	<i>Process:</i> EcDsDistributionServer <i>Classes:</i> DsDdConfiguration DsDdPfConfigFile	Configuration control information (queues, thresholds, suspend and resume status) is sent to the database via GUI or command line.
Set Configuration Info	One per scheduling request	<i>Libraries:</i> DsDdSSH, DsDdC <i>Classes:</i> DsDdRequestMgrBaseC, DsDdConfiguration, DsDdDistRequestS	<i>Process:</i> EcDsDdistGui <i>Classes:</i> DsDdConfiguration DsDdPfConfigFile	Current configuration information from the Science Data Servers is sent to the Sybase database and accessed by the Operations staff for expedient data distribution or product order distribution scheduling.
Cancel	One per priority cancel request	<i>Library:</i> DsDdC <i>Class:</i> DsDdRequestMgrBaseC	<i>Process:</i> EcDsDdistGUI <i>Class:</i> DsGuiDistRequest	Enables the Operations staff with the capability to cancel the priority of a request.
Resume	One per resume request	<i>Libraries:</i> DsDdB, DsDdC, DsDdl <i>Class:</i> DsDdRequestMgrC	<i>Process:</i> EcDsDdistGUI <i>Class:</i> DsGuiDistRequest	Enables the Operations staff with the capability to resume requests when most errors are encountered.
Suspend	One per suspend request	<i>Libraries:</i> DsDdB, DsDdC, DsDdl <i>Class:</i> DsDdRequestMgrC	<i>Process:</i> EcDsDdistGUI <i>Class:</i> DsGuiDistRequest	Enables the Operations staff with the capability to suspend a request(s).
Set Priority	One per priority change	<i>Library:</i> DsDdS <i>Classes:</i> DsDdMedia, DsDdRequestListS, DsDdRequestMgrReal, DsDdDistRequestS	<i>Process:</i> EcDsDdistGUI <i>Class:</i> DsGuiDistRequest	Enables the Operations staff with the capability to change the priority of a distribution request.

**Table 4.1.2.5-1. DDIST CSCI Process Interface Events (3 of 4)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Mark Shipped	One per tape request	<i>Libraries:</i> DsDdB, DsDdC, DsDdl <i>Class:</i> DsDdRequestMgrC, DsDdMedia	<i>Process:</i> EcDsDdistGUI <i>Class:</i> DsGuiDistRequest	Enables the Operations staff with the capability to mark the tape request for shipment via GUI.
Status & Error Information	One per Distribution Request	<i>Library:</i> MsAcCIntSh <i>Class:</i> EcAcOrderCMgr	<i>Process:</i> EcDsDistributionServer <i>Classes:</i> DsDdMedia, DsDdBaseQueue	Distribution Request status and error information is obtained by the Operations staff from the EcDsDistributionServer.
Data Packing Info. (E-mail)	One per distributed data location	<i>Library:</i> DsDdSSH <i>Class:</i> DsDdMedia	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia	Notification sent to the user via e-mail with location of data to distribute or has been distributed.
Push Data Instructions	One per distribution request	<i>Libraries:</i> DsDdB, DsDdC, DsDdl <i>Classes:</i> DsDdRequestMgrC, DsDdGranuleC	<i>Process:</i> EcDsScienceDataServer	The EcDsScienceDataServer assembles instructions to send data to the EcDsDistributionServer. The EcDsDistributionServer sends a request to the EcDsStArchiveServer to push the data, via the FTP service, followed by a signal file to the destination specified in an acquire instruction.
Print packing slips	As many needed per distribution request	<i>Library:</i> DsStPrinter <i>Class:</i> DsStPrinter	<i>Process:</i> EcDsStDistributionServer <i>Class:</i> DsDsReadytoShipQueue	The EcDsStDistributionServer tells the EcDsStPrintServer to print a file on a given staging disk. The EcDsStDistributionServer requests the EcDsStPrintServer to print out a packing list file associated with a distribution request after the request has been completed successfully.

**Table 4.1.2.5-1. DDIST CSCI Process Interface Events (4 of 4)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request Management Services	Upon initial system start-up or per CSCI request.	<i>Library:</i> EcAgInstrm <i>Classes:</i> EcAgManager, MsAgRegistry	<i>Process:</i> EcMsAgSubAgent	The EcMsAgSubAgent provides a basic management library of services to the processes, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li> <b>Lifecycle commands</b> - The EcMsAgSubAgent forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.                             </li> </ul> The MCI CSCI also interfaces with other CSCIs or CSCs to perform the following: <p><b>User Profile Request</b> – The MCI CSCI provides requesting CSCIs or CSCs with access to user profile information such as e-mail address and shipping address to support their processing activities.</p>

#### 4.1.2.6 Data Distribution Data Stores

Table 4.1.2.6-1 provides descriptions of the individual DDIST CSCI data stores entitled collectively “DDIST Data Stores” in the DDIST CSCI architecture diagram. More details on these database tables can be found in the Data Distribution Database Design and Schema Specifications for the ECS Project.

**Table 4.1.2.6-1. DDIST CSCI Data Stores**

<b>Data Store</b>	<b>Type</b>	<b>Description</b>
DsDdFile	Sybase	This data store holds the distribution files maintained/processed by the EcDsDistributionServer. Table Abbreviation "F" is used as the standard naming convention for stored procedures.
DsDdGranule	Sybase	This data store holds the distribution granules maintained/processed by the EcDsDistributionServer. Table Abbreviation "G" is used as the standard naming convention for stored procedures.
DsDdParameterList	Sybase	This data store holds the GIParameter list for each request maintained/processed by the EcDsDistributionServer. The EcDsScienceDataServer provides data from external metadata (i.e., via the MCF). Request information is initiated here first. Table abbreviation "PL" is used as the standard naming convention for stored procedures.
DsDdRequest	Sybase	This data store holds the distribution requests maintained/processed by the EcDsDistributionServer. Table abbreviation "R" is used as the standard naming convention for stored procedures.

### **4.1.3 Storage Management Software Description**

#### **4.1.3.1 Storage Management Functional Overview**

The Storage Management (STMGT) CSCI stores/archives, manages, and retrieves non-document earth science data and provides a user friendly graphical user interface (GUI) for operations. The STMGT CSCI manages all physical storage resources for all the DSS CSCIs and processes including: tape robotic archive, RAID disk cache, on-line storage, and peripheral devices used for ingesting data from and distributing data to physical media such as various tape sizes or drive types.

The STMGT CSCI manages both long-term high capacity archival of data (data repository) and short term/temporary storage (working storage/cache management). The STMGT CSCI controls associated file access services to the archive, handles short-term data storage needs for the INGST CSCI, the DDIST CSCI, the SDSRV CSCI, and the PRONG CSCI. The STMGT CSCI also provides access to physical media peripheral devices for both the INS and the DDIST CSCI.

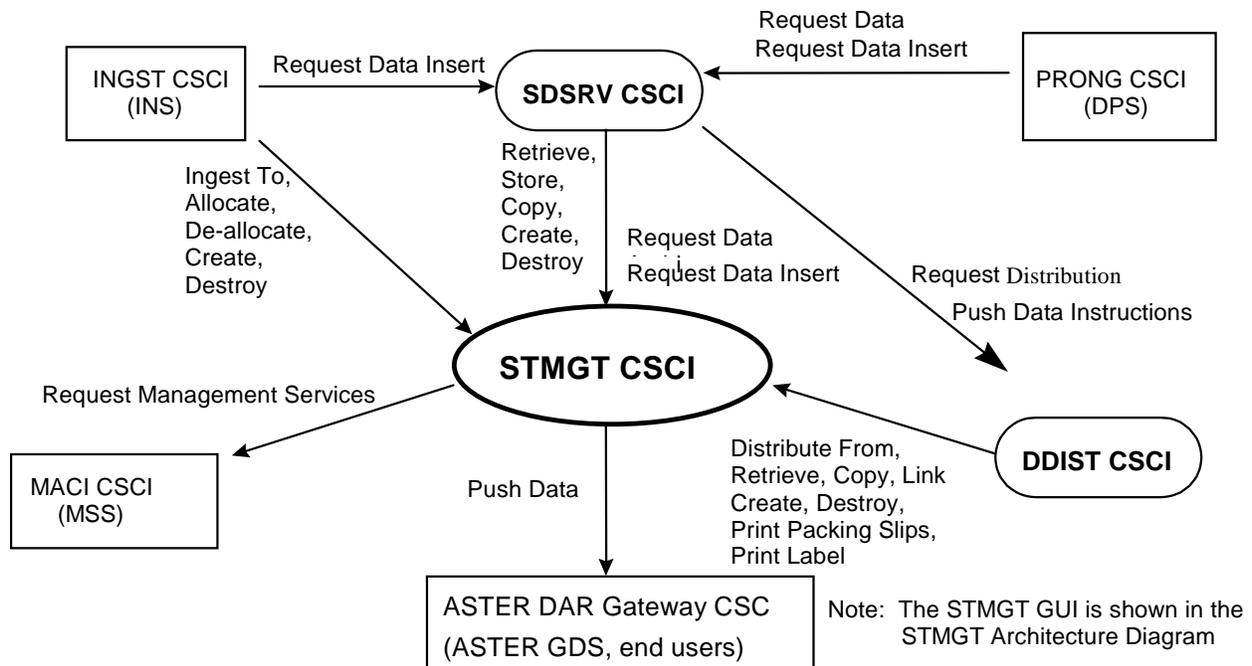
During data ingest, the STMGT CSCI provides interfaces to enable ingest and obtain access to disk space, FTP services, and shared resource peripheral devices. The STMGT CSCI copies files into the archive for permanent storage. During data distribution, the SDSRV CSCI and the DDIST CSCI copy files from the archive and allocates magnetic disk space for staging the files. The DDIST CSCI also allocates the peripheral devices shared with the INGST CSCI for copying of files to hard media, or to copy files for electronic distribution. The STMGT CSCI maintains a user pull area to allow for electronic pull distribution.

The STMGT CSCI provides retrieval and storage methods to the DDIST CSCI and the SDSRV CSCI to support storing and providing data for their client CSCIs. The PRONG CSCI is a client CSCI in both storage and retrieval requests by retrieving lower level data via FTP Push Acquire through the SDSRV and DDIST CSCIs and by storing data via insert request to the SDSRV CSCI. In addition to the product files created by the PRONG CSCI, the following types of files are stored for the PRONG CSCI: intermediate product files (interim files), production history files, metadata files, and lower level data files such as raw science data. The STMGT CSCI stores files as a result of insert requests to the SDSRV CSCI by the INGST CSCI and the PRONG CSCI. The STMGT CSCI retrieves files from the archive to satisfy sub-setting requests submitted to the SDSRV CSCI and by acquire requests submitted to the SDSRV CSCI and routed through the DDIST CSCI.

The STMGT GUI provides a simple and consistent interface to set various system parameters, identify errors, analyze the underlying problem(s), and develop corrective measures. The persistence of the data is maintained in a database through a STMGT GUI interface. The GUI provides a method to manage system resources such as servers, cache thresholds, on-line storage availability, and peripherals. The GUI also provides the operator with the capability to track the status of files, tapes, and drives for an enhanced level of quality control.

#### 4.1.3.2 Storage Management Context

Figure 4.1.3.2-1 is the STMGT CSCI context diagram. The diagram shows the events sent to other ECS subsystems and events received from other ECS subsystems. The STMGT CSCI has direct interfaces with the INGST and PRONG CSCIs.



**Figure 4.1.3.2-1. STMGT CSCI Context Diagram**

Table 4.1.3.2-1 provides descriptions of the interface events shown in the STMGT CSCI context diagram.

**Table 4.1.3.2-1. STMGT CSCI Interface Events (1 of 2)**

Event	Interface Event Description
Request Data Archive	The SDSRV CSCI sends archive requests to the STMGT CSCI archive server for various categories of data. The archive services are essentially identical for all data categories and consist of request to retrieve files, store files, copy files, or create/destroy staging disk temporary files.
Allocate	Requests are sent from the INGST CSCI to the STMGT CSCI for allocating peripheral devices for ingesting and distribution of data.
Distribute From	Enables the copying of files from staging to an allocated peripheral resource.
Ingest To	Enables the copying of files from peripheral resource to staging areas.
De-allocate	Requests are sent from the INGST CSCI to the STMGT CSCI for de-allocating previously allocated resources.
Retrieve	Requests are sent from the SDSRV CSCI to the STMGT CSCI for the retrieval of data or products from archival storage.
Store	Requests are sent from the INGST CSCI to the STMGT CSCI for data insertion (for archiving purposes) into the science data server inventory/archives.
Create	Enables users to allocate area on shared staging disks.
Copy	Enables users of staging disks to copy between and within staging disks.
Link	Enables users of staging disks to link files from read-only staging disks to their current disk.
Destroy	Enables users of staging disks to destroy their reference to staging and free disk space for future use.
Print Packing Slips	The DDIST CSCI sends a request to the STMGT CSCI to print the packing list associated with the distribution request.
Request Data	For the STMGT CSCI, the PRONG CSCI sends requests to the SDSRV CSCI for a particular data granule to be pushed onto the DPS science processor, via the FTP service, to be used as input for PGE processing or for SSIT work.
Request Data Insert	For the STMGT CSCI, the PRONG CSCI sends requests to the SDSRV CSCI for a particular file or files be inserted into the archive, and their metadata be catalogued into the SDSRV inventory, as a granule of a particular ESDT short name and version. These files can be processing output, static files received with PGEs, PGE Tape Archive (TAR) files, Aps, SSAPs or DAPs, failed PGE tar files, or production history files. The INGST CSCI sends requests to the SDSRV CSCI to insert data into the data inventory including metadata.
Request Management Services	<p>The MACI CSCI provide a basic management library of services to the CSCIs, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include:</p> <ul style="list-style-type: none"> <li> <b>Lifecycle commands</b> – The MACI CSCI forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run. </li> </ul>

**Table 4.1.3.2-1. STMGT CSCI Interface Events (2 of 2)**

Event	Interface Event Description
Push Data	The STMGT CSCI pushes data (i.e., EDS), via the FTP service, to the DAR Communications Gateway for data distribution per user request. A signal file is also sent to indicate the completion of the file transfer.
Insert Data	The INGST CSCI sends an insert request to the SDSRV CSCI for data received from external providers.
Request Distribution	The SDSRV CSCI sends distribution requests to the DDIST CSCI for various categories of data. The distribution services on those data are essentially identical for all data categories. In response, data packaging information is sent back to the user via e-mail.
Push Data Instructions	The SDSRV CSCI assembles instructions to send data to users, sends the instructions to the DDIST CSCI and the DDIST CSCI sends a request to the STMGT CSCI FTP Distribution server to push the data followed by a signal file to the destination specified in an acquire instruction.
Print Label	A label is printed for distribution of physical media.

### 4.1.3.3 Storage Management Architecture

Figure 4.1.3.3-1 is the STMGT CSCI architecture diagram. The diagram shows the events sent to the STMGT CSCI processes and the events the STMGT CSCI processes sends to other processes or gateways for remote systems such as ASTER GDS. Table 4.1.3.4-1 provides descriptions of the interface events shown in the STMGT CSCI architecture diagram.

The STGMT CSCI is composed of the following processes:

- EcDsStArchiveServer, EcDsStmgtGui (used for archiving data)
- EcDsStStagingMonitorServer, EcDsStStagingDiskServer, EcDsStmgtGui (used for staging data)
- EcDsSt8MMServer, EcDsStD3Server, EcDsStIngestFtpServer, EcDsStFtpDisServer, EcDsStPrintServer, EcDsStmgtGui (used for resource management)
- EcDsStPullMonitorServer (used to monitor data transfers)

The STMGT GUI software is an interface with the database to set parameters and configurations to control the servers.

The Archival Management and Storage System (AMASS) is an automated library management system. See the AMASS R Overview by EMASS, version 4.9.1, dated August 1998, document number 600705 for more information about the AMASS management and storage system.



**Table 4.1.3.4-1. STMGT CSCI Processes**

Process	Type	COTS/ Developed	Functionality
EcDsStArchiveServer	Server	Developed	An Archive Server provides access to stored data. There can be multiple archive servers running at a given site, each with its own type of data or storage media. Distinctions between Archive Servers are based on the Archive Id as part of the metadata for each ESDT. For requests retrieving files from the Archive, the staging disk is located on the same host as the Archive. For data being inserted, based on network architecture, move files to staging disk located on a node with the appropriate network access. The server id can be configured differently based on an ESDT in the SDSRV and INGEST CSCIs for flexibility.
EcDsStStagingMonitorServer EcDsStStagingDiskServer	Servers	Developed	The Staging Monitor Server manages the group of data files retrieved from the archive and placed into a cache area on a staging disk. A list of these data files is maintained so subsequent data retrieval requests are fulfilled immediately without requiring an additional archive access. The Staging Monitor Server also deletes files with zero access to prevent the cache area from becoming too full. The STMGT CSCI supports multiple staging monitors.  The Staging Disk Server manages shared disk space. The Staging Disk Server enables disk space allocations and file reservations between staging directories and from non-staging to staging directories. The STMGT software supports multiple instances of the Staging Disk Server.
(ResourceManager) EcDsSt8MMServer, EcDsStD3Server, EcDsStIngestFtpServer, EcDsStFtpDisServer, EcDsStPrintServer	Servers	Developed	The Resource Manager process schedules access to shared peripheral devices. Queues are maintained for requests based on priority and request receipt time. Each type of resource pool has its own resource manager. Individual resource manager processes identified are 8-mm tape, D3Tape, IngestFtp, DistributionFtp, and printer.
EcDsStPullMonitorServer	Server	Developed	The Pull Monitor Server manages the files in the user pull area. As files are retrieved (i.e., electronically pulled) from the user pull area by respective ECS users or as the files become stale (their time-out periods have expired), the Pull Monitor Server deletes them. Several algorithms are available for monitoring and maintaining the data levels at a specified capacity.
AMASS	Server	COTS	Provides a Unix File System interface to the robotics to control the media where data is written and read.
EcDsStmgmtGUI	GUI	Developed	This GUI enables the Operations staff to store and retrieve information from Sybase tables during the Ingest process for data storage or distribution.

### 4.1.3.5 Storage Management Process Interface Descriptions

Table 4.1.3.5-1 provides descriptions of the interface events shown in the STMGT CSCI architecture diagram.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (1 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Allocate	One per allocation	<i>Library:</i> DsStRes <i>Class:</i> DsStResourceProvider	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia <i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Provides the capability to allocate peripheral devices for ingesting and distribution of data.
Distribute From	One per data copy from staging	<i>Library:</i> DsStTape <i>Class:</i> DsStTape	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules	Enables the copying of files from staging to an allocated peripheral resource.
Ingest To	One per data copy from peripheral	<i>Library:</i> DsStTape <i>Class:</i> DsStTape	<i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Enables the copying of files from peripheral resource to staging.
InsertFile	One per file insert into the Pull cache	<i>Library:</i> DsStPull <i>Class:</i> DsStPullMonitor	<i>Process:</i> EcDsStFtpDsServer <i>Class:</i> DsStDistributionFtp	Used by the FtpDistribution server to insert a file into the Pull cache. The return value from the Pull Monitor server indicates whether the file needs to be transferred to the cache or is already resident.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (2 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Deallocate	One per de-allocation	<i>Class:</i> DsStResourceProvider	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia <i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Provides the capability to de-allocate previously allocated resources.
Copy file	One per file copy	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingMonitor	<i>Process:</i> EcDsStArchiveServer <i>Class:</i> DsStArchiveReal	This is a Unix file copy out of the AMASS cache by buffered read/write software.
Request Data	One per retrieval request	<i>Library:</i> DsStSt <i>Class:</i> DsStArchive	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDbInterface <i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules	Provides the capability to request the retrieval of data objects from archival storage.
Request Data Insert	One per insertion request	<i>Library:</i> DsStSt <i>Class:</i> DsStArchive	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsGeESDT	Provides the capability to request the insertion of data objects into archival storage.
Copy	One per file copy	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingDisk	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules <i>Process:</i> EcDsScienceDataServer <i>Library:</i> DsGe <i>Class:</i> DsGeESDT	Enables users of staging disks to copy between and within staging disks.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (3 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Create	One per area allocation	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingDisk	<i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF <i>Process:</i> EcDsScienceDataServer <i>Library:</i> DsGe <i>Class:</i> DsGeESDT <i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia	Enables data providers to allocate an area on shared staging disks.
Link	One per link file	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingDisk	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules <i>Process:</i> EcDsScienceDataServer <i>Library:</i> DsGe <i>Class:</i> DsGeESDT	Enables users of staging disks to link files from read-only staging disks to their current disk.
Destroy	One per destruction	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingDisk	<i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Enables users of staging disks to destroy their reference to staging.
Push Data	One per acquire instruction	<i>Library:</i> DsStDisFtp <i>Class:</i> DsStDistributionFtp	<i>Process:</i> EcDsDistributionServer <i>Classes:</i> DsDdScheduler DsDdMedia	The EcDsDistributionServer receives instructions to distribute data and the EcDsStFtpDisServer FTP pushes the data followed by a signal file to the destination specified in an acquire instruction.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (4 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Get Capacity	One per maximum capacity determination	<i>Library:</i> DsStRes <i>Class:</i> DsStResourceProvider	<i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdMedia	Data Distribution determines the effective maximum capacity of a media type. This is used to determine the number of media needed to satisfy a given request.
Mount Tape	One per physical application of tape	<i>Library:</i> DsStTape <i>Class:</i> DsStTape	<i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Enables the Operations staff to attach and load tapes to hardware peripherals for reading or writing to the tapes.
Un-mount Tape	One per physical application of tape	<i>Library:</i> DsStTape <i>Class:</i> DsStTape	<i>Process:</i> EcInGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	Enables the Operations staff to unload and detach tapes from hardware peripherals after reading or writing to the tapes.
DownAccess Count	One per request for access count decrease	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingMonitor	<i>Process:</i> EcDsStStagingDiskServer <i>Class:</i> DsStStagingDiskReal	Enables a user to request that the Staging Monitor decrease the access count to a given file. This is done after a file has been copied or a link to a file in the cache has been removed.
Staging Table	One per staging table activity	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingTable	<i>Process:</i> EcDsStStagingDiskServer <i>Class:</i> DsStStagingManager	The staging table is a list of currently allocated staging directories.
Staging Table Entry	One per staging table entry	<i>Library:</i> DsStSt <i>Class:</i> DsStStagingTable	<i>Process:</i> EcDsStStagingDiskServer <i>Class:</i> DsStStagingManager	This is an entry in the Staging Table. It contains the size allocated for the staging disk, the access permitted, the directory identifier, the request id, and the owner of the disk.
MkDir	One per directory creation	<i>Library:</i> DsStPull <i>Class:</i> DsStPullMonitor	<i>Process:</i> EcDsSrFreDisServer <i>Class:</i> DsStDistributionFtp	Creates the directory on the User Pull area to which files are linked and from which the external requester can pull files.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (5 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request Management Services	Upon initial system start-up or per process request.	<i>Library:</i> EcAgInstrm <i>Classes:</i> EcAgManager, MsAgRegistry	<i>Process:</i> EcMsAgSubAgent	The EcMsAgSubAgent provides a basic management library of services to the processes, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The EcMsAgSubAgent forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>
Push Data Instructions	One per distribution request	<i>Libraries:</i> DsDdB, DsDdC, DsDdI <i>Classes:</i> DsDdRequestMgrC, DsDdGranuleC	<i>Process:</i> EcDsScienceDataServer	The EcDsScienceDataServer assembles instructions to send data and sends the instructions to the EcDsDistributionServer. The EcDsDistributionServer sends a request to the EcDsStFtpDisServer to push data, via the FTP service, followed by a signal file to the destination specified in an acquire instruction.

**Table 4.1.3.5-1. STMGT CSCI Process Interface Events (6 of 6)**

Event	Event Frequency	Interface	Initiated By	Event Description
Get Capacity	One per maximum capacity determination	<i>Library:</i> DsStRes <i>Class:</i> DsStResourceProvider	<i>Process:</i> EcDsDistributionServer <i>Classes:</i> DsDdScheduler	EcDsDistributionServer determines the effective maximum capacity of a media type. This is used to determine the number of media needed to satisfy a given request.
Distribution FtpPull	One per pull request	<i>Library:</i> DsStPull	<i>Process:</i> EcDsDistributionServer	The EcDsDistributionServer sends a request to the EcDsStFtpDisServer to check for the presence of a file or move a file to the cache area.
Retrieve	One per retrieval request	<i>Library:</i> DsStSt <i>Class:</i> DsStArchive	<i>Process:</i> EcDsScienceDataServer <i>Class:</i> DsDbInterface <i>Process:</i> EcDsDistributionServer <i>Class:</i> DsDdGranules	Provides the capability to request the retrieval of data objects from archival storage.
Schedule	One per media	<i>Library:</i> DsStRes	<i>Processes:</i> EcDsSt8MMServer, EcDsStD3Server	Uses tables in the database to determine if a drive is available to handle the request.
Print Packing Slips	One per distribution request	<i>Library:</i> DsStPrinter <i>Class:</i> DsStPrinter	<i>Process:</i> EcDsStDistributionServer <i>Class:</i> DsDsReadyToShipQueue	Tells the EcDsStPrintServer to print a file on a given staging disk. The EcDsStDistributionServer only prints out the packing list file associated with a distribution request after the request has completed successfully.

#### 4.1.3.6 Storage Management Data Stores

Table 4.1.3.6-1 provides descriptions of the individual data stores (entitled collectively “Database”) in the STMGT CSCI architecture diagram. More detailed information on these database tables can be found in the CDRL 311 document on Storage Management.

**Table 4.1.3.6-1. STMGT CSCI Data Stores (1 of 3)**

<b>Data Store</b>	<b>Type</b>	<b>Description</b>
DsStDevice	Sybase	The data store (database table) to contain an entry for each peripheral device the STMGT CSCI uses to service requests to ingest or distribute data. Since this data store is a look-up for entries in the Request entity, this data is initialized prior to software installation.
DsStBackup	Sybase	Stores a reference to every file currently being backed-up related to Archive Backup and Restore functionality.
DsStBackupHistory	Sybase	Stores a history of every file successfully backed-up related to Archive Backup and Restore functionality.
DsStCache	Sybase	This data store identifies every instance of a cache related to Pull Monitor Cache Management. The current default is for only one existing cache.
DsStCacheDirectory	Sybase	This data store contains an entry for each directory with a Storage Management cache. An entry is inserted into the data store every time a new cache is created, or a directory is added to an existing cache.
DsStCacheFile	Sybase	This data store contains an entry for each file Storage Management Pull Monitor is currently processing (originally implemented via the pull_list table). An entry is inserted into the entity for each file retrieved from the Archival Management And Storage System (AMASS). DsStFileLocation (originally implemented via the pull_link table) tracks the individual cache locations of (or links to) the file.
DsStConfigParameter	Sybase	This data store contains an entry for information to configure and initialize each DsStServerType supported by Storage Management. The data consists of information currently accessed through configuration files (*.CFG) plus information as it pertains to the status and node of operation for each server. An entry is inserted for each parameter that a server uses.  Two types of parameters can be defined within the constructs of the DsStConfigParameter table, startup and run-time. Startup parameters require the associated server be restarted in order for the parameters to be used. Run-time parameters can be changed without restarting the server (i.e., the server periodically queries the configuration table for new values).
DsStDeleteLogCache File	Sybase	This data store contains a historic record of each file the STMGT CSCI deletes from its (Pull Monitor) cache (or the DsStCacheFile table). This entity maintains a history of file usage and cache usage for reporting and analysis purposes. An entry is inserted into the entity via a delete trigger in the DsStCacheFile table.
DsStErrorAttribute	Sybase	This data store is required for the DsStErrorDetails class. This data store provides a mapping between character mnemonics and numeric error codes. It defines the attributes for each error, and provides adequate characterization for appropriate retry/recovery procedures from the error attributes.
DsStErrorText	Sybase	This data store provides text descriptions and suggested recovery actions for each error code, and presents errors in a meaningful manner.

**Table 4.1.3.6-1. STMGT CSCI Data Stores (2 of 3)**

Data Store	Type	Description
DsStEventLog	Sybase	<p>This data store contains a history of events and COTS errors encountered by the STMGT CSCI. The STMGT CSCI inserts records into the table using the stored procedure DsStELInsert.sp. The calling sequence is: DsStELInsert @EventNumber=value, @EventMessage=value, @EventDate=value, @EventType=value.</p> <p>The STMGT CSCI inserts a new ERROR_LOG entry each time an event occurs or an error is encountered. The Operations staff has the capability to purge this entity periodically based on a date/time value.</p>
DsStFileLocation	Sybase	<p>This data store maintains the location(s), in a cache, of each file recorded in the DsStCacheFile table. As each file is retrieved from the archive and written to one of the Storage Management caches, an entry is inserted in the entity. The entry must correspond to an existing DsStCacheFile entry.</p> <p>Upon insertion of a DsStFileLocation record, the corresponding DsStCacheFile record is updated by incrementing or decrementing the ActiveLinkCount via a Trigger and setting the LastAccessDate to the current system date and time.</p>
DsStNextId	Sybase	<p>This data store is used to track the automatic, sequential generation of unique identifying keys for various tables in the Storage Management database (DsStErrorLog, DsStDeleteLogCacheFile, DsStBackupHistory, DsStRestoreHistory, and DsStConfigParameter.TimerId)</p>
DsStOffsite	Sybase	<p>This data store is used for mapping the off-site identifiers (e.g., GSF, ERC, ...) with their appropriate Cell location (i.e., gsfc.nasa.gov) and identifying the specified Archive directory for (remote) backup.</p>
DsStSchedule	Sybase	<p>This data store contains an entry for schedule requests of a peripheral device the Storage Management receives. An entry is inserted into the entity every time a peripheral device is reserved to service a request to ingest or distribute data. The unit of measure for the estimated duration is minutes.</p>
DsStServerType	Sybase	<p>This data store contains all types of servers administered and configured by Storage Management and their associated descriptions. 8 types of standard servers are currently pre-populated with the database construction due to no user interface currently existing to administer (e.g., Archive, Staging Monitor, Pull Monitor, Distribution FTP, 8MM Tape, Staging Disk, Ingest FTP, D3,...)</p>
DsStVolumeGroup	Sybase	<p>This data store contains 'volume group' information (the section of the ECS Archive being accessed) from configuration files such as the path currently pointed to and a history of paths related ONLY to a specific Archive server type.</p>
DsStSlot	Sybase	<p>This data store contains an entry for each peripheral stacker device's slots the EcDsStArchiveServer and EcDsDistributionServer use to service ingest or data distribution requests, respectively. Since this data store is a look-up entity for entries in the Request entity, this data is initialized prior to software installation.</p>

**Table 4.1.3.6-1. STMGT CSCI Data Stores (3 of 3)**

<b>Data Store</b>	<b>Type</b>	<b>Description</b>
DsStStacker	Sybase	This data store contains an entry for each peripheral stacker the STMGT CSCI uses to service ingest or data distribution requests. Since this data store is a look-up entity for entries in the Request entity, this data is initialized prior to software installation.
DsStStagingDisk	Sybase	Stores information (e.g., staging disk size, owner, disk number, and priority request id) about requests for staging disk services.
DsStArchiveRequest	Sybase	Stores all requests for archive services until they are completed.
DsStDistributedFile	Sybase	Contains an entry for each file STGMT is currently processing related to a DsStFtpRequest entry.
DsStFile	Sybase	Contains an entry for each file STGMT is currently processing related to a DsStArchiveRequest entry.
DsStFtpRequest	Sybase	Stores information (e.g., pull directory name and parameter list for FTP request) about FTP specific requests.
DsStGenericRequest	Sybase	Stores common information to all STMGT requests (e.g., checkpoint state, request status, and type of operation, and owner name) regardless of type.
DsStRestore	Sybase	Stores information needed for the restoration from backup for files in the archive.
DsStRestoreHistory	Sybase	Historical records of all archive file restoration activities.
DsStStgMonRequest	Sybase	Contains information (e.g., unique file name, original file name, file size, and compression type) for Staging Monitor specific requests.
DsStVersion	Sybase	Contains version information about the installed database. Data includes the date of installation, the version number of the database installed, and the latest version number available for the loaded database.

#### **4.1.4 Data Server Subsystem Hardware**

##### **4.1.4.1 Access Control and Management Hardware CI Description**

The Access Control and Management HWCI (ACMHW), provides access to the Data Server subsystem for subsystem and direct “push/pull” user access, provides tools and capabilities for system administration, and supports the infrastructure of the Data Server. This HWCI controls logical data server access, maintains sessions, provides sub-setting support, directs service requests to other appropriate Data Server Subsystem configuration items, and supports the control and data flow for electronic distributions. The Access Control and Management hardware is logically divided into the Administration Stations (AS) and the Access/Process Coordinators (APCs). The number, type, and configuration of ASs and APCs depend on site requirements and the number of data servers supported. Table 4.1.4.1-1 provides descriptions of the ACMHW.

**Table 4.1.4.1-1. Access Control and Management HWCI Descriptions**

Server Name	Class/Type	Description
Administration Stations	OPS Workstation	SUN workstations
Access/Process Coordination	APC Data Server	Two SGI Servers with Multi-processor Capabilities and a SUN Server

Administration Stations (AS) hosts provide access to the Administration Services for one or more data servers. These services provide Data Server Administrators with the capability to modify and monitor the configuration of the data server. The data server configuration includes resource availability, number and location of items, data server schema, advertised services, data types and archiving strategy. In addition, this set of services includes the capability to perform Archive maintenance functions. The ASs are mid-sized workstations executing GUI packages to accommodate the imposed I/O and processing requirements.

The Access/Process Coordinators (APCs) are the interface to the data server services. The APCs support session establishment, management, and control. and are used to access the data server services. Sessions management (from a user perspective) provides a variety of data server resources including results sets, cached compute-on-demand data objects, search contexts, etc. These resources impose a computational load and an I/O load on the APC processors. A greater I/O load is imposed by the APCs functioning as a data throughput mechanism for electronic data distribution in Release 4 and for data manipulation via sub-setting in Release 4. APC platforms also support the ingest, storage, and distribution of the Browse data. The Browse data hardware accommodates the different DAAC sizes. At the smaller DAACs, Browse data resides on a RAID disk, at larger sites, like GSFC, Browse data is stored in the EMASS Automated Media Library (AML) robotic libraries handling optical media for HP optical drives. Each APC consists of four platforms configured in pairs as follows: each pair is an SGI host acting as a server and running a Science Data Server database and a SUN host running the ECS Science Data Server custom code. One pair is primary, while the other is the standby. The APCs provide the computer resources, search engines, and tools to operate on data retrieved from repositories. Algorithms operating on site data can execute on an APC or a computer in the Data Processing Subsystem.

The APC SGI server host accommodates electronic ingest of Browse data and data from sources external to the DAAC, including electronic distribution, and manages the requests to the Data Server. APC storage supports functions such as sub-setting, storing user session context, keeping track of user session interactions that can be suspended or resumed, and the Browse collection.

Two identical AS workstations supply AS fail-over/recovery capability via redundant operation. Either workstation can be used independently as an operator workstation and in the event of a failure of one, the other can assume the full role. The two APC servers in a standby configuration share a common disk pool. This disk pool is configured as an outboard stack of RAID disk drives dual-ported to the two servers. One of the servers is configured as primary, the other as a standby secondary. The active server maintains the current system configuration files on disk. In some cases, both servers are active with separate primary responsibilities and shared secondary responsibilities. The servers can accommodate degraded mode fail-over capability in the event of

a failure. When a failure on a primary APC is detected, the operator initiates a fail-over procedure. During a fail-over, the backup APC server reads the system status files from the shared disk and begins operating as primary.

Three types of network failures can affect the DSS. If the FDDI cable between a host and the FDDI concentrator were damaged, a new cable would be required. No other configuration would be required. If an individual port on the FDDI concentrator fails, the attached host must be moved to another port, and should be replaced quickly. Note the above failures result in a service interruption only to the workstations. Since all servers/processors are attached to two hubs, they can communicate as normal in the event of a cable or concentrator fault, and the applications are unaffected by the event.

Recovery/fail-over for HWCIs is described in the 920-TDx-014 series of documents. There is a version for each DAAC, indicated by the letter appearing in place of the “x.” The document provides the recovery procedure for each host.

#### **4.1.4.2 Data Repositories Hardware CI Description**

Data Repositories (DRs) HWCI (DRPHW) are hardware to store and maintain data permanently. Different technologies are used to instantiate DRs depending on the volume and type of data to be stored, the access patterns of the data, and additional unique requirements imposed on the repository (i.e., data maintenance requirements, backup and restore functions, media management and control, etc.).

DRs are classified as “permanent”, meaning the services to monitor and maintain data integrity for large data holdings are supported by this repository's storage technology. A copy of all data at a site not considered temporary is eventually maintained in a site permanent DR.

The File and Storage Management System (FSMS) host platform is an SGI machine. In DRPHW CIs there are at least two such hosts to serve as a primary and a backup. At the sites with more than two FSMS hosts, one of the hosts serves as a backup to all other hosts and the WKSHW host if any (See Section 4.1.4.3 for the WKSHW Description).

The Data Server's servers and workstations are directly connected to the DAAC FDDI network and HIPPI fabric. The Data Server processors/servers contain dual-attached station (DAS) cards, dual-homed to separate FDDI concentrators. This provides redundancy so full connectivity exists to the servers even in the event of a concentrator failure. The workstations contain single-attached station (SAS) cards and each are connected to a single concentrator, and they are also split across concentrators so they are not all connected to the same unit. The FDDI concentrators are in turn connected to the FDDI switch.

Table 4.1.4.2-1 provides descriptions for the Data Repository HWCI.

**Table 4.1.4.2-1. Data Repository HWCI Descriptions**

Item Name	Class/Type	Description
Data Repository	Archive Robotics	STK Powderhorn or EMASS AML-E, site dependent
	Magnetic Tape Drives	STK Redwood or IBM Magstar, site dependent
	SMP Server Workstation (FSMS Host)	SGI Server with Multi-processor Capabilities.

#### 4.1.4.3 Working Storage Hardware CI Description

Working Storage HWCI (WKSHW) of the Data Server supplies storage for temporary file and buffer storage within the Data Server architecture. This pool contains the interim processing data. WKSHW provides the staging capacity for data acquires and inserts of the interim data.

The logical configuration of the WKSHW is consistent at all DAAC sites. The physical WKSHW accommodates sites with a variety of data storage capacities and data rates supported.

All interim product data used by the Data Processing Subsystem are staged in the WKSHW for a maximum period of 90 days (most interim products are far more short-lived). At most sites, disk storage is supplied for the interim products. Only at EDC, where interim product data accumulation is very significant, archival robotics and tapes are used for this storage as a second tier of WKSHW. A separate AMASS supports the functioning of the robotic WKSHW. Table 4.1.4.3-1 provides descriptions for the Working Storage HWCI.

**Table 4.1.4.3-1. Working Storage HWCI Descriptions**

Item Name	Class/Type	Comments
Working Storage Primary Tier	RAID (host attached)	
Working Storage Secondary Tier	Tape Robotics	At applicable sites

Using a RAID for storage implementation provides fault tolerance of the WKSHW. The RAID provides degraded mode of operations for a single disk failure. Second tier storage RMA is equivalent to the rest of the robotic library storage in the archive repository (See Section 4.1.4.2 for the Data Repository Hardware Description). The WKSHW host is backed up by a standby in the DRPHW configuration.

#### 4.1.4.4 Distribution and Ingest Peripheral Management Hardware CI Description

The hardware of the Distribution and Ingest Peripheral Management (DIPHW) supports the media distribution methods for data dissemination from the system and hard media ingest of data into the system. Hard media distribution and ingest is used in an assortment of data recording peripherals. Robotic control is used where applicable to minimize operator involvement.

Data distributed from the data server is buffered for a full day shift (8 hours at most sites). The buffering of the data is provided by the Distribution Storage Management hardware in the data server.

The distribution and ingest peripheral hardware supply the hard media for inter-site, user data distribution scheduling and management. This HWCI contains media drives, jukeboxes/stackers, server hosts, and disk storage for network distribution.

Staging disks in the Distribution and Ingest configuration items of the DSS serve as a buffer to the data pull process. The buffered data pull optimizes performance of the data retrieval for distribution or data processing/archiving after ingest.

The Distribution and Ingest Management hardware includes SUN server controlled recording devices for both hard media data distribution and hard media data ingest.

**Equipment complement:** RAID (host attached), 8-mm tape drives and stackers, 3480/3490 drives, CD Recordable, printers, optical scanner, FAX, two SUN servers in a load sharing configuration.

## 4.2 Ingest Subsystem Overview

The Ingest Subsystem (INS) ingests data into ECS repositories in accordance with approved ICDs. Data is accepted from a variety of external data providers in a variety of formats predefined within ECS regarding the expected metadata and metadata characteristics. The INS supports a variety of ingest requests and preprocesses the data for archiving into ECS. The preprocessing depends on the attributes of the ingested data such as data type, data format, and the level to which the ingested data has been processed from raw instrument data.

The data types are formally referred to as Earth Science Data Types (ESDTs). An ESDT is a defined data set associated with a given mission/instrument or identified grouping and is registered in an ECS ESDT Baseline List. The ESDT is identified with a short name, a long name, a collection description, and information on file type, metadata, formats, and services provided such as sub-setting by a given attribute. An example of an ESDT is AST\_L1A, the short name for the ASTER reconstructed Level 1A data set, unprocessed instrument digital counts with radiometric (LR) and geometric (LG) coefficients attached. ESDTs can also be ancillary data, algorithms, correlative, and calibration data.

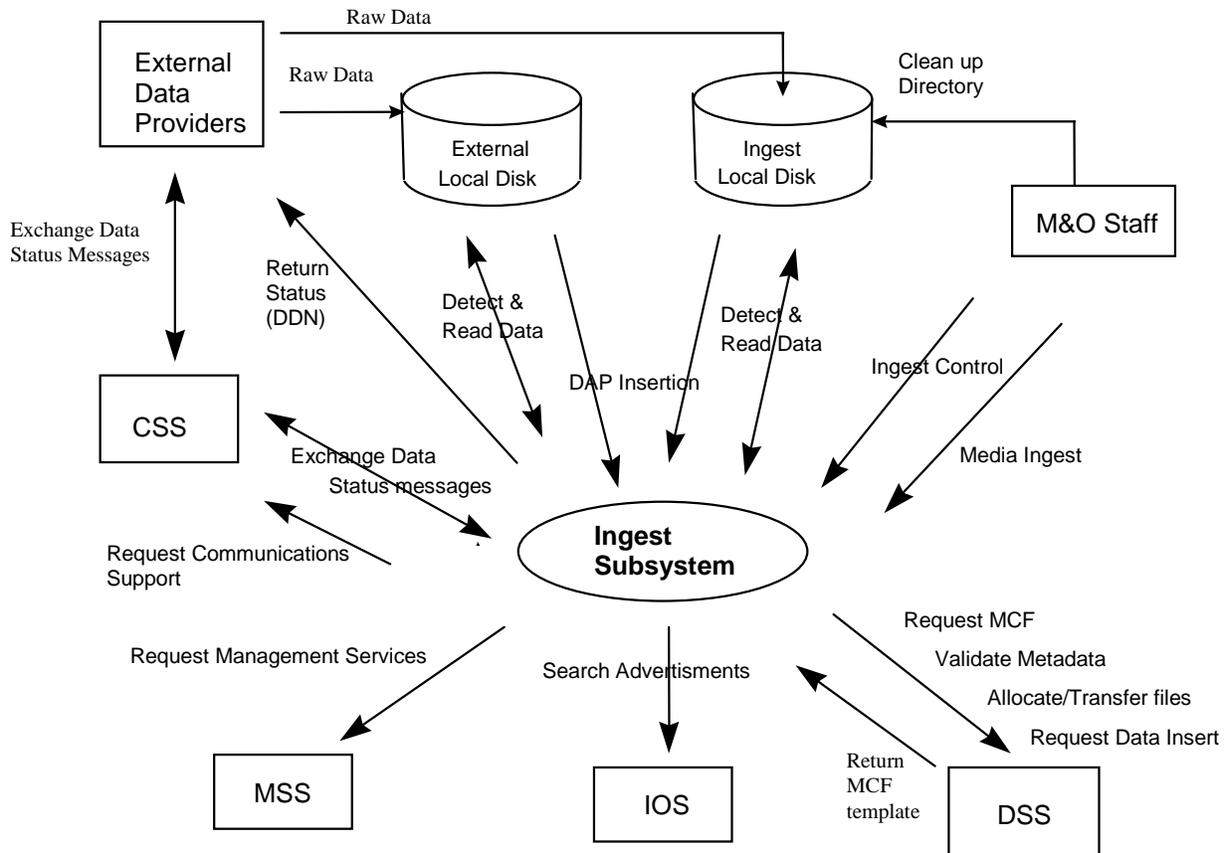
The INS software processes execute in a specific pattern based on the ESDT being ingested. The processes include an automated ingest process, a polling ingest process, and a media ingest process. Ingest processes provide for the receipt of external data, which is archived within the ECS Science Data Processing Segment (SDPS) archival system. Specific ingest process procedures are established to support each unique INS interface and allow the processing interface parameters to be modified as interface and mission requirements evolve. For a given incoming data set corresponding to an ESDT, the INS performs data preprocessing, metadata extraction, and directs the DSS SDSRV CSCI to perform metadata validation.

Data is staged (prepared for transfer) to one of two areas depending on the data level, ESDT, and other data set specific characteristics:

- Level 0 (L0) data received from external data providers and other selected data (EDOS ancillary data) is staged to the INS working storage area. Metadata is extracted and quality checked on the working storage area. The L0 data is transferred to an archive data repository in the DSS for long-term storage.
- Non-L0 data (such as non-EDOS ancillary data and L1A - L4 data from external facilities) is staged directly to the working storage area in the DSS. Extraction of metadata and quality checking are performed on the data by the INS software residing in the DSS processor hardware. The non-L0 data is transferred to a DSS archive data repository for long-term storage.

### Ingest Subsystem Context

Figure 4.2-1 is the INS context diagram. The diagram shows the events sent to the INS from other ECS subsystems and the events the INS sends to other ECS subsystems, the Operations staff, and external providers. Table 4.2-1 provides descriptions of the interface events shown in the Ingest Subsystem context diagram.



**Figure 4.2-1. Ingest Subsystem Context Diagram**

**Table 4.2-1. Ingest Subsystem Interface Events (1 of 2)**

Event	Interface Event Description
Exchange Data Status Messages	Data status messages are sent to and from the CSS Gateways via Remote Procedure Calls (RPCs). A Data Availability Notice (DAN) is sent to the INS and afterwards additional data status messages are exchanged with the INS.
Return Status (DDN)	The INS returns the status of a request received from an External Data Provider by transmitting a Data Delivery Notice (DDN) to the External Data Provider directly and to the CSS Gateway for the Landsat 7 automated ingest process.
Request MCF	The INS requests the Metadata Configuration File (MCF) template, from the DSS, for a particular ESDT short name prior to a data insert request.
Return MCF template	The template is sent to the INS that populates the file to create a MCF.
Validate Metadata	The DSS validates the metadata files that the INS has populated.
Raw data	The raw data (L0) provided from the external provider to the ECS. Some external providers put this data on an external local disk for ECS to pull while others push the data onto a local INS internal disk.

**Table 4.2-1. Ingest Subsystem Interface Events (2 of 2)**

Event	Interface Event Description
Request Communications Support	The CSS provides a library of services available to each ECS subsystem. The services required to perform the specific subsystem assignments are requested by the subsystem from the CSS. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Detect & Read Data	The INS polls for data files or Delivery Record files in an agreed upon location (External to ECS or Ingest internal Local Disk).
Clean up Directory	The Operations staff sends clean up instructions to the Ingest Local Disk for data clean up. Currently, delete and remove Unix commands are executed from the Unix command line to clean up the Ingest local disk.
DAP Insertion	Delivered Algorithm Packages (DAPs) are located on a Local Disk (external or internal to a DAAC) and are inserted into the ECS via the automated polling ingest interface.
Ingest Control	The Operator controls the Ingest function by monitoring requests, changing database parameters, viewing history, and performing manual media ingest via a GUI.
Allocate/Transfer files	The INS accesses and allocates/de-allocates staging disks and transfers data files into the DSS. De-allocation of files deletes data not marked as persistent data.
Request Data Insert	The INS sends requests to the DSS to insert a particular file or files into the archive. Inserted data is accompanied by metadata. The metadata is catalogued in the SDSRV inventory as a granule of a particular ESDT short name and version.
Search Advertisements	The Interoperability Subsystem (IOS) receives requests to search for advertisements and returns Universal References (URs) for ECS managed objects (hosts, servers, routers, other devices). The INS uses the advertisement information to locate the relevant data servers to interact with.
Media Ingest	The M&O staff provide instructions via a GUI to ingest data from hard media supplied from an ECS user.
Request Management Services	<p>The MSS provides a basic management library of services to the subsystems, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include:</p> <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The MSS forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>

## Ingest Subsystem Structure

The INS is one CSCI and one HWCI. Ingest backup hardware runs in the Test Mode as long as there are no failures. The configuration items are:

- Ingest (INGST) CSCI provides the software capability to acquire data by various methods and transfers the data into the ECS. These methods include an Automated transfer from prior request information, Polling with or without Delivery Records for data placed at predetermined locations, and a Media transfer method which includes reading tapes. The INGST CSCI also stores and manages request information, and provides for data preprocessing storage and insertion into the appropriate ECS storage location.
- Ingest Client HWCI (ICLHW) supports INGST in bringing data into the ECS from an external interface. This HWCI also includes an Ingest Workstation for execution of the Ingest GUI.

## Use of COTS in the Ingest Subsystem

- Rogue Wave's Tools.h++

The Tools.h++ class libraries provide basic functions and objects. These libraries must be installed with the INS software for any of the INS processes to run.

- Rogue Wave's DBTools.h++

The DBTools.h++ class libraries interact with the Sybase SQL database server. These libraries must be installed with the INS software to interact with Sybase. The INS uses an interface software process control file (PCF) to obtain access to the Sybase Server.

- Rogue Wave's Net.h++

This is a C++ class library to provide an object-oriented interface to Inter-Process Communication (IPC) and network communication services. The Net.h++ framework enables developed code to be portable to multiple operating systems and network services. These libraries must be installed with the INS software to support the interface to other subsystems.

- Integrated Computer Solutions' (ICS) Builder Xcessory (built on X/Motif)

The Builder Xcessory GUI builder tool modifies the displays of Ingest GUIs. The tool also generates the C++ code to produce Ingest GUIs at run time. There is no operational part of the Builder Xcessory needed at run-time.

- Sybase (xaClient, SQL Server)

This set of Sybase products provides a relational database to store INS related information and must be installed on the platform where INS software resides.

- Tivoli Client

This product provides the interface to monitor the INS software for system administration purposes. Although Tivoli Client is part of MSS, this COTS product must be installed on the platform where the INS resides for Ingest status to be available at the system level.

- DCE Client

This product provides the communications between INS and other subsystems. DCE can reside on one or both sides of the interface and must be installed on the platform where the INS resides. Although the DCE Client is part of the CSS, this COTS product must be installed on the platform where the INS software resides for INS to run in the ECS operational and test environments.

- DCE Driver

This product provides the communications to support a Personal Computer (PC) environment. The DCE Driver must be installed on the computer where MS Access is installed.. Although the DCE Driver is part of the CSS, this COTS product must be installed on the platform where the INS software resides for INS to run.

- MS Access

This product provides an interface to authorized users in a PC environment for updating specific database values and must be installed on the computer where MS Access is used.

- HP OpenView Client

This product provides system administration capabilities to control the INS (startup and shutdown). The HP OpenView Client must be installed on the platform where the INS software resides. Although HP OpenView Client is part of the MSS, this COTS product must be installed on the platform where the INS resides for INS to run.

- UNIX Network Services

DNS, NFS, E-mail/Bulletin Board, FTP, TCP/IP and the other Unix services provided are obtained from the CSS and are described in section 4.8 of this document.

## **4.2.1 INGST Software Description**

### **4.2.1.1 INGST Functional Overview**

The INGST CSCI supports a variety of external interfaces. The application-level protocol set up for data transfer is potentially different for each of the external interfaces. As a result, a separate ingest software application is required to facilitate data transfer for each interface. To minimize the software development effort and make it easier to accommodate new external interfaces in the future, data ingest from external interfaces is categorized, based on common characteristics and ingest processes.

### **1) Automated Network Ingest Interface**

Automated Network Ingest occurs when a Data Availability Notice (DAN) is supplied to the ECS. ECS receives the DAN and schedules automated network data transfer from the source. The DAN describes the location of the available data. External data providers are responsible for developing application software to interact with ECS automated network ingest software.

### **2) Automated Polling Ingest Interface**

- Polling with Delivery Record -- ECS periodically checks an agreed-upon network location for a Delivery Record file. The Delivery Record file contains information identical to that in a DAN. The Delivery Record describes the location of the available data. The data location could be on a working storage device within ECS, where an external data provider has previously transferred the data.
- Polling without Delivery Record -- ECS periodically checks an agreed-upon network location for available data. All data in the location make up a collection of ingest data of one specific ESDT, with one file per data granule.

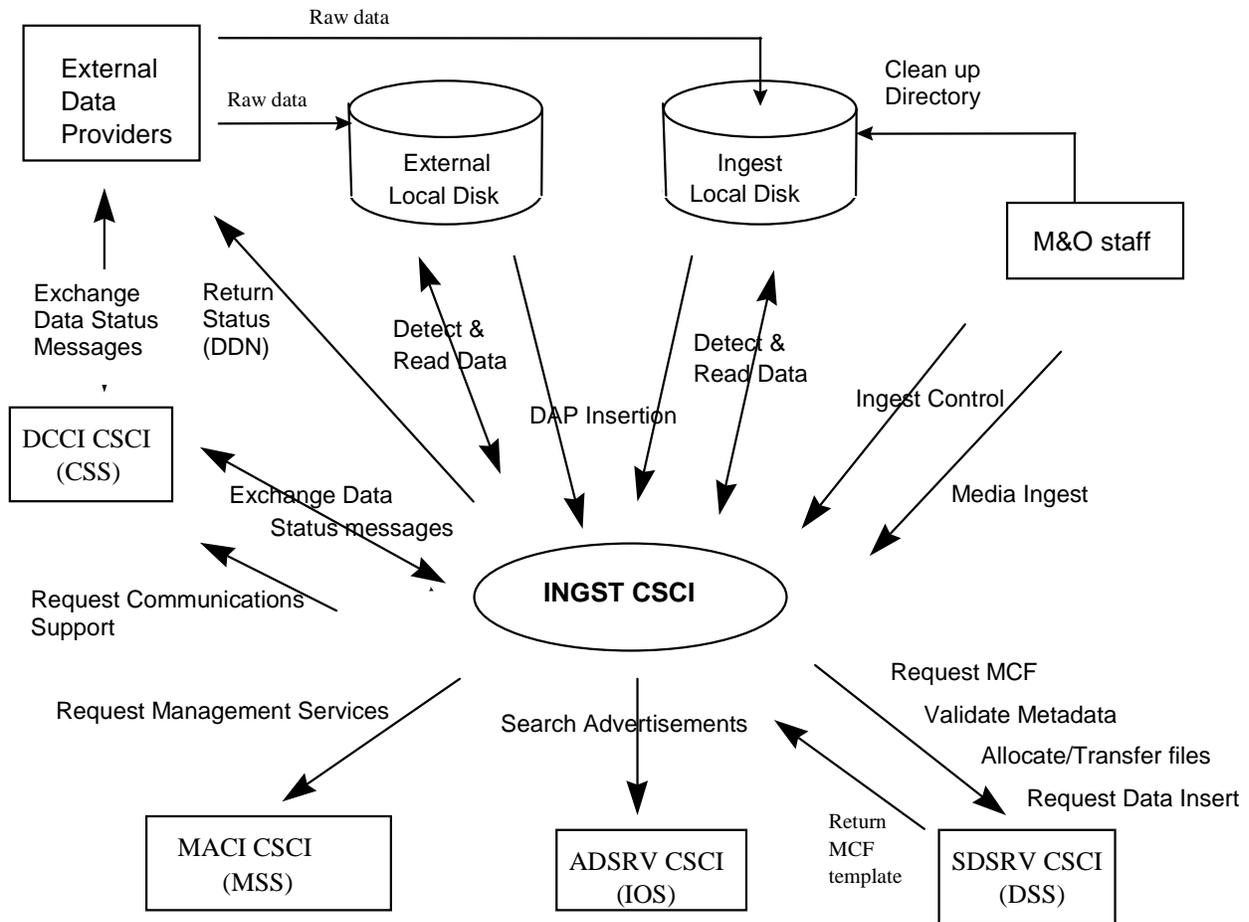
### **3) Manual Media Ingest Interface (via a GUI)**

Manual data transfer mechanisms -- data can be transferred from physical media. Physical Media Ingest enables authorized institutions or science users to provide data on hard media and provides a backup procedure for facilities when automated network data transfer is temporarily unavailable. The hard media must contain information identical to the Delivery Records described above, in a standard file format, or the data provider must separately provide Delivery Records to a specified ECS location in the standard file format. Hard media data transfer involves data transfer from one of several ingest peripheral types found at a DAAC.

The INGEST CSCI includes the processes for ingesting data as described and provides a process for managing requests, and for inserting granule data into the ECS.

#### **4.2.1.2 INGEST Context**

Figure 4.2.1.2-1 is the INGEST CSCI context diagram. The diagram shows the events sent to the INGEST CSCI and the events the INGEST CSCI sends to other CSCIs. Table 4.2.1.2-1 provides descriptions of the interface events shown in the INGEST CSCI context diagram.



**Figure 4.2.1.2-1. INGST CSCI Context Diagram**

**Table 4.2.1.2-1. INGST CSCI Interface Events (1 of 2)**

Event	Interface Event Description
Exchange Data Status Messages	Data status messages are sent to and from DCCI CSCI via Remote Procedure Calls (RPCs). A Data Availability Notice (DAN) is sent to the INGST CSCI and afterwards additional data status messages are exchanged with the INS.
Return Status (DDN)	The INGST CSCI returns the status of a request received from an External Data Provider by transmitting a Data Delivery Notice (DDN) to the External Data Provider directly and to the CSS Landsat 7 Gateway via the network.
Request MCF	The INGSTCSCI requests the MCF template, from the SDRSV CSCI, for a particular ESDT short name prior to a data insert request.
Return MCF template	The template is sent to the INGST CSCI that populates the file to create a MCF.

**Table 4.2.1.2-1. INGST CSCI Interface Events (2 of 2)**

Event	Interface Event Description
Validate Metadata	The SDSRV CSCI validates the metadata files that the INGST CSCI has populated.
Raw data	The raw data (L0) provided from the external provider to the ECS. Some external providers put this data on an external local disk for ECS to pull while others push the data onto a local INGST CSCI internal disk.
Request Communications Support	The DCCI CSCI provides a library of services available to each ECS subsystem. The services required to perform the specific subsystem assignments are requested by the subsystem from the CSS. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Detect & Read Data	The INGST CSCI polls for data files or Delivery Record files in an agreed upon location (External to ECS or Ingest internal Local Disk).
Clean up Directory	The Operations staff sends clean up instructions to the Ingest Local Disk for data clean up. Currently, delete and remove Unix commands are executed from the Unix command line to clean up the Ingest local disk.
DAP Insertion	Delivered Algorithm Packages (DAPs) are located on a Local Disk (external or internal to a DAAC) and are inserted into the ECS via the automated polling ingest interface.
Ingest Control	The Operator controls the Ingest function by monitoring requests, changing database parameters, viewing history, and performing manual media ingest via a GUI.
Allocate/Transfer files	The INGST CSCI accesses and allocates/de-allocates files and transfers data files into the SDSRV archives. De-allocation of files deletes data not marked as persistent data.
Request Data Insert	The INGST CSCI sends insert requests, to the SDSRV CSCI, for a particular file or files to be inserted into the archives. Inserted data is accompanied by metadata. The metadata is catalogued in the SDSRV inventory (archives) as a granule of a particular ESDT short name and version.
Search Advertisements	The ADSRV CSCI receives requests to search for subscription event advertisements and returns Universal References (URs) for ECS managed objects (hosts, servers, routers, other devices). The INS uses the advertisement information to locate the relevant data servers to interact with.
Media Ingest	The M&O staff provide instructions via a GUI to ingest data from hard media supplied from an ECS user.
Request Management Services	<p>The MACI CSCI provides a basic management library of services to the CSCIs, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include:</p> <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The MACI CSCI forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>

### 4.2.1.3 INGST Architecture

The Automated Network Ingest Interface (EcInAuto) sets up ingest sessions with external data providers, like Landsat 7 LPS, via the CSS Gateways. External data providers submit Data Availability Notices (DANs) to request a data ingest session. The Polling Ingest Interface (EcInPolling) polls accessible file system locations to detect data to be ingested. This process submits an equivalent DAN or the information for INGST to create a DAN. The Media Ingest Interface (EcInGUI) enables authorized science users or institutions to submit a DAN and the data to be ingested via physical media.

The Automated Network Interface, the Polling Ingest Interface, and the Media Ingest all submit ingest requests (containing DANs) to the Ingest Request Manager (EcInReqMgr) and the EcInReqMgr submits data granule requests to the Ingest Granule Server (EcInGran). The EcInGran manages subsequent request processing. The EcInGran invokes a Data Transfer task to transfer data from external locations. The EcInGran also invokes a data preprocessing task to preprocess ingested data (e.g., process metadata and validate metadata parameters) and invokes the Data Server Insertion Task to insert data into the Data Server.

Figure 4.2.1.3-1 is the INGST CSCI architecture diagram. The diagram shows the events sent to the INGST CSCI processes and the events the INGST CSCI processes send to other processes.

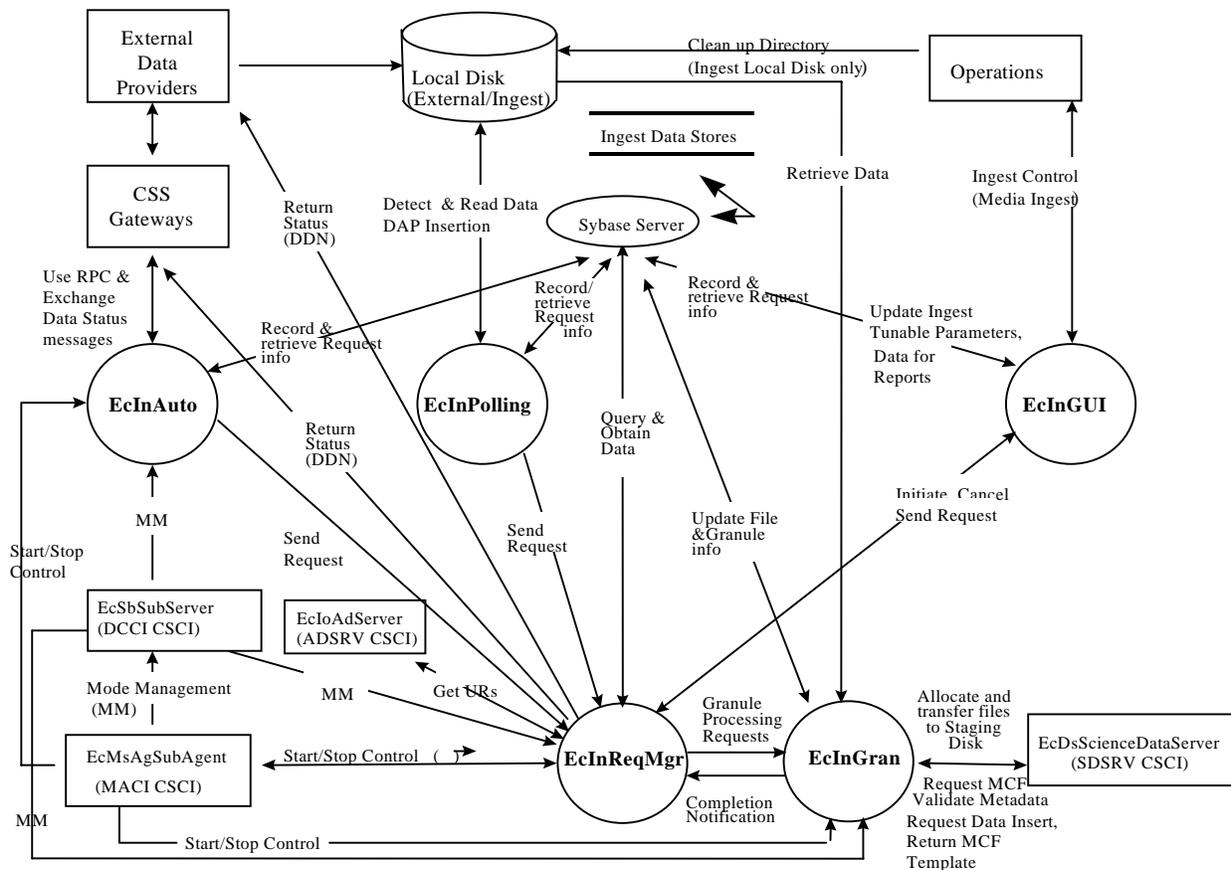


Figure 4.2.1.3-1. INGST CSCI Architecture Diagram

#### 4.2.1.4 INGST Process Descriptions

Table 4.2.1.4-1 provides the descriptions of the processes shown in the INGST CSCI architecture diagram.

**Table 4.2.1.4-1. INGST CSCI Processes (1 of 2)**

Process	Type	Source	Functionality
EclnAuto	Server	Developed	<p>Provides fundamental capabilities for data ingest into the ECS, upon receipt of a DAN. This process can be tailored for a specific interface. RPCs are used to request ingest services to schedule data transfer from the source. EclnAuto also</p> <ul style="list-style-type: none"> <li>• manages single requests at a time</li> <li>• sends the requests to the EclnRegMgr for processing</li> <li>• checks DAN information</li> <li>• invokes an RPC to the Request Manager to begin request processing</li> <li>• sends and receives data status messages</li> <li>• cleans up request information.</li> </ul>
EclnPolling	Client	Developed	<ul style="list-style-type: none"> <li>• Creates the appropriate polling request</li> <li>• detects new files of interest at tunable periods of time in either external or local disk locations (by checking an agreed upon network location for available data)</li> <li>• creates a unique identifier for the request</li> <li>• submits requests</li> <li>• reports the status of the ongoing requests</li> </ul>
EclnGUI	GUI	Developed	<p>Provides Operations personnel the capability, via GUI Interface,</p> <ul style="list-style-type: none"> <li>• to perform physical media ingest (to ingest data from hard media)</li> <li>• to monitor the ingest history log, to monitor the status of ongoing ingest requests</li> <li>• to initiate or cancel an ingest request and to modify ingest configuration parameters</li> </ul>
EclnReqMgr	Server	Developed	<ul style="list-style-type: none"> <li>• Manages the ingest request traffic and the processing of the ingest requests, and</li> <li>• provides the capability to process multiple ingest requests concurrently by placing the request in a queue when a certain threshold (number of requests) has been reached.</li> <li>• In the event of a failure, the EclnReqMgr process restores on-going requests from the Ingest database.</li> </ul>

**Table 4.2.1.4-1. INGST CSCI Processes (2 of 2)**

Process	Type	Source	Functionality
EclnGran	Server	Developed	<p>Provides services to perform the required data preprocessing and the subsequent insertion of the data into the appropriate Data Server. The preprocessing of data consists of:</p> <ul style="list-style-type: none"> <li>• converting the data (if needed)</li> <li>• extracting the metadata into the standard ECS metadata format (if needed)</li> <li>• performing required metadata existence and parameter range checks</li> <li>• updating the metadata with ingest specific metadata (e.g., start and stop date/time for ingest)</li> </ul> <p>EclnGran tracks the ingest requests and coordinates the ingest processing including:</p> <ul style="list-style-type: none"> <li>• transferring data</li> <li>• performing data preprocessing</li> <li>• sending an insertion request to the appropriate Data Server</li> <li>• updates the granule state</li> <li>• transfers data files into Ingest</li> <li>• builds file lists</li> <li>• groups files with a valid ESDT</li> </ul>
Sybase	Server	COTS	<p>Stores and provides access to the INS internal data. In particular, the database stores the Ingest operations databases -- Ingest History Logs and the Ingest request checkpoint state, and template information. See Section 4.2.1.6 INGST Data Stores.</p>

#### 4.2.1.5 INGST Process Interface Descriptions

Table 4.2.1.5-1 provides descriptions of the interface events shown in the INGST CSCI architecture diagram.

**Table 4.2.1.5-1. INGST CSCI Process Interface Events (1 of 5)**

Event	Event Frequency	Interface	Initiated By	Event Description
Use RPC & Exchange Data Status messages	One per RPC from CSS	<i>Class:</i> InAutoIngestIF_1_0_Mgr	<i>Process:</i> EcCsLandsat7 Gateway	The EclnReqMgr initiates a session with the CSS Gateways by invoking a Remote Procedure Call (RPC). DAN messages are sent from the Gateway, and DAA messages are sent to the Gateway.

**Table 4.2.1.5-1. INGST CSCI Process Interface Events (2 of 5)**

Event	Event Frequency	Interface	Initiated By	Event Description
Detect and read Data	One per poll from EclnPolling	Various ID directories on various local disks	<i>Class:</i> InPollingIngest Session	The EclnPolling polls for data files or Delivery Record files in an agreed location (Local Disk).
Retrieve Data	One per Science Data retrieval	FTP Pull/ Directory on local disk with information	<i>Process:</i> EclnGran	The EclnGran retrieves Science Data for a given ESDT from the Local Disk (external or Ingest Local Disk).
Clean up Directory	One per Unix command to delete or remove	Directories on local disk	Operator/ Unix command	The Operator sends clean up instructions to the Ingest Local Disk for data clean up. The Unix commands for delete and remove are executed from the Unix command line to clean up the Ingest Local Disk.
Record & retrieve Request info	One per request	Ingest database	<i>Library:</i> InDBaccUt	Requests are recorded into the Ingest database for reference and are a source for restarts and re-initializations of outstanding requests.
Send Request	One per request to Request Manager	<i>Class:</i> InRequestManager	<i>Library:</i> InGuiUt <i>Classes:</i> InAutoNtwkIngestMgr, InPollingIngestSession, InMediaIngestRPUtil	Processing requests from one of the three ingest processes (EclnAuto, EclnPolling, and EclnGUI) are sent to the EclnRegMgr.
DAP Insertion	One per DAP insertion	Local disk	<i>Class:</i> InPollingIngestSession	Delivered Algorithm Packages (DAPs) are located at the Local Disk for insertion into the ECS with the Ingest Polling process.

**Table 4.2.1.5-1. INGST CSCI Process Interface Events (3 of 5)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Ingest Control	One per Ingest Operation	Ingest database or InRequestManager or Operations	Operator <i>Library:</i> InGuiUt <i>Classes:</i> InRequestControllerRPUtil, InMediaIngestRPUtil, InHistoryLogRPUtil, InOperatorToolsRPUtil	The Operator controls the Ingest function by monitoring requests, changing database parameters, viewing history, and performing manual media ingest via a GUI.
Initiate or Cancel	One per update request	<i>Class:</i> InRequestManager	<i>Process:</i> EclnGUI <i>Library:</i> InGuiUt <i>Classes:</i> InRequestControllerRPUtil, InMediaIngestRPUtil	The EclnGUI is an interface to the EclnReqMgr process to update requests by initiating or canceling a request.
Update File & Granule info	One per file or granule update	Ingest database	<i>Process:</i> EclnGran <i>Library:</i> InDBaccUt <i>Classes:</i> InRequestProcessData, InRequestFileInfo	Updated file and granule information is obtained from the Ingest database by the EclnGran process.
Update Ingest Tunable Parameters	One per update of stored parameters	Ingest database	<i>Process:</i> EclnGUI <i>Library:</i> InGuiUt <i>Class:</i> InOperatorToolsRPUtil	The EclnGUI is the interface to the Ingest database to update the stored parameters affecting the functions of the EclnAuto, EclnPolling, EclnReqMgr, and EclnGran processes.
Data for Reports	One per data request	Ingest database	<i>Process:</i> EclnGUI <i>Library:</i> InGuiUt <i>Class:</i> InHistoryLogRPUtil	The EclnGUI obtains data from the Ingest database to generate reports.

**Table 4.2.1.5-1. INGST CSCI Process Interface Events (4 of 5)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Return Status (DDN)	One per status of	External Data Provider EcCsLandsat7Gateway	<i>Process:</i> EclnReqMgr <i>Class:</i> InRequest	The EclnReqMgr returns status of a request to the External Providers via a Data Delivery Notice and Landsat 7status to the CSS Gateway.
Query & Obtain Data	One per query of Ingest Database	Ingest database	<i>Process:</i> EclnReqMgr <i>Library:</i> InDBaccUt	The EclnReqMgr queries the Ingest database for ingest processing information and obtains data from the database.
Granule Processing Requests	One per granule processing request	<i>Class:</i> InGranuleAsync_S	<i>Process:</i> EclnReqMgr <i>Class:</i> InRequest	The EclnReqMgr sends processing requests to the EclnGran process for granule processing.
Completion Notification	One per granule completion	<i>Library:</i> InGranuleC <i>Class:</i> InGranuleAsync_C	<i>Process:</i> EclnGran <i>Class:</i> InGranuleAsync_S	The EclnGran process sends a completion notification to the EclnReqMgr when a granule for a request is completed.
Request Data Insert	One per insert into the archive	<i>Library:</i> DsCI <i>Class:</i> DsCIRequest	<i>Process:</i> EclnGran <i>Library:</i> InPreprocess <i>Class:</i> InDataServerInsertionTask	The EclnGran process requests a file or files to be inserted into the SDSRV archives, and the associated metadata is catalogued in the SDSRV inventory (archives), as a granule of a particular ESDT short name and version.
Allocate and transfer files to Staging Disk	One per Science Data file activity	<i>Libraries:</i> DsStSt, DsStRes <i>Classes:</i> DsStStagingDisk, DsStResourceProvider	<i>Process:</i> EclnGran <i>Library:</i> InResource <i>Class:</i> InResourceIF	The EclnGran process accesses, allocates, de-allocates, and transfers Science Data files to the EcDsStStagingDisk. De-allocation deletes the data not marked as persistent data.
Request MCF	One per access of MCF	<i>Library:</i> DsCI <i>Class:</i> DsCIDescriptor	<i>Process:</i> EclnGran <i>Library:</i> InPreprocess <i>Class:</i> InDataPreprocessTask	The EclnGran process requests the MCF template from the EcDsScienceDataServer.

**Table 4.2.1.5-1. INGST CSCI Process Interface Events (5 of 5)**

<b>Event</b>	<b>Event Frequency</b>	<b>Interface</b>	<b>Initiated By</b>	<b>Event Description</b>
Return MCF Template	One per MCF template populated	<i>Library:</i> DsCI <i>Class:</i> DsCIDescriptor	<i>Process:</i> EcInGran <i>Library:</i> InPreprocess <i>Class:</i> InDataPreprocess Task	The EcInGran populates the MCF template, and returns the populated MCF template to the EcDsScienceDataServer.
Validate Metadata	One per metadata validation	<i>Library:</i> DsCI <i>Class:</i> DsCIDescriptor	<i>Process:</i> EcInGran <i>Library:</i> InPreprocess <i>Class:</i> InDataPreprocess Task	The EcInGran process requests the EcDsScienceDataServer to perform a validation of the metadata files.
Get URs (Search Advertisements)	One per use of Universal References	<i>Library:</i> InAdSearch <i>Class:</i> IoAdApprovedAdvSearch Command	<i>Process:</i> EcInReqMgr <i>Library:</i> InUpdateUR <i>Class:</i> InUpdateUR	The EcInReqMgr receives Universal References from the EcIoAdServer as a result of an advertisement request. Ingest uses the Advertisement Information to locate the relevant Data Server to contact for data requests.
Start/Stop Control (Request Management Services)	One per start or stop system instruction	<i>Classes:</i> InAutoNtwkIngestMgr, InRequestServer, InGranuleServer	<i>Process:</i> EcMsAgSubAgent	The EcMsAgSubAgent process sends start and stop commands to begin or terminate the EcInAuto, EcInReqMgr, and the EcInGran processes.
Mode Management (MM) (Request Communications Support)	One per Mode Management Message	<i>Classes:</i> InAutoNtwkIngestMgr, InRequestMgr, InGranuleServer	<i>Process:</i> EcMsAgAgent	The EcSbSubServer issues mode management information (which it receives from the EcMsAgSubAgent) to the EcInAuto, EcInReqMgr, and EcInGran processes.

#### 4.2.1.6 INGST Data Stores

The INGST CSCI uses the COTS product Sybase to store related INGST information on a physical medium. The stored information is divided into four functional areas:

1. Checkpoint and reactivate ingest processing
2. Summary or historical information for collecting and reporting metrics

3. Ingest configuration (e.g., thresholds) and template information
4. Validation tables for the INS GUI and software

Table 4.2.1.6-1 provides descriptions of the individual data stores used by the INGST CSCI. The architecture diagram shows a single data store entitled “Ingest Data Stores” for simplification.

**Table 4.2.1.6-1. INGST CSCI Data Stores (1 of 2)**

<b>Data Store</b>	<b>Type</b>	<b>Description</b>
InRequestProcessHeader	Sybase	Provides checkpoint storage of ingest request processing information associated with a given ingest request. Upon request process completion, copies of these records are stored in InRequestSummaryHeader data store and the request processing ingest information is deleted.
InRequestProcessData	Sybase	Provides checkpoint storage of data granule processing information associated with a given ingest request. Upon request process completion, copies of these records are stored in the InRequestSummaryData data store and the granule processing data information is deleted.
InRequestFileInfo	Sybase	Provides checkpoint storage of file information associated with a data granule within a given ingest request.
InRequestSummaryHeader	Sybase	Provides long-term storage of summary request-level statistics associated with a given ingest request. Summary records are copied upon ingest request processing completion and the processing records are deleted from the system.
InRequestSummaryData	Sybase	Provides long-term storage of summary data type statistics associated with a given data granule in a given ingest request. Summary records are copied upon ingest request processing completion and the processing records are deleted from the system.
InSourceMCF	Sybase	Initially, pre-populated with the valid metadata types for each file type. It “points” to the metadata and indicates “how” to handle the data in a standard object description language (ODL) format.
InSystemParameters	Sybase	Stores current system thresholds that limit ingest request traffic and data volume.
InExternalDataProviderInfo	Sybase	Stores thresholds on ingest request traffic and data volume for External Data Providers.
InValGranuleServerUR	Sybase	Provides validation for the Granule Server Universal References (URs).
InGranuleServerInfo	Sybase	Stores thresholds on granule traffic and data volume for each Granule Server.
InDataTypeTemplate	Sybase	Initially, pre-populated with current, valid Earth Science Data Types (ESDTs) that the INS is capable of ingesting. A MSAccess97 front-end interface exists to maintain the data in the data store.
InESDTMap	Sybase	Stores the information for special mappings of the ESDT and the Client data type.
InFileTypeTemplate	Sybase	Initially, pre-populated with all valid file types that make up an ESDT. A MSAccess97 front-end interface exists to maintain the data in the data store.

**Table 4.2.1.6-1. INGST CSCI Data Stores (2 of 2)**

<b>Data Store</b>	<b>Type</b>	<b>Description</b>
InMediaType	Sybase	Stores the valid values of the media type available that can be ingested.
InNextAvailableID	Sybase	Stores the next available RequestID to be given.
InValDataGranuleState	Sybase	Stores all the valid values for a data granule state.
InValIngestType	Sybase	Stores all the valid values for an ingest type.
InValNotifyType	Sybase	Stores all the valid values for a notify type.
InValParameterClass	Sybase	Initially, pre-populated with all the valid values for a parameter class.
InValRequestState	Sybase	Stores all the valid values for a request state.
InEDPAddressMap	Sybase	Initially, pre-populated with the IP address for an External Data Provider.

## **4.2.2 Ingest Subsystem Hardware**

### **4.2.2.1 Ingest Client Hardware CI Description**

The Ingest Client HWCI (ICLHW) Server accommodates the required ingest volumes, including I/O, and processing capabilities to support internal data transfers associated with metadata validation and extraction, and to transfer data to the Data Server or Data Processing Subsystem. The disks are sized to accommodate the functionality and provide contingency space for the transfer of more than one day's worth of data within a 24-hour period.

The Ingest Server is a 64-bit SGI machine. Dual processors are installed on the Ingest Server (See 920-TDx-001 series of base-line documents).

The Ingest Server is configured with at least 256 MB of memory with one-way interleaving to support the processors.

The SGI architecture provides configurable I/O subsystems that attach to the backplane. These I/O subsystem items are referred to as IO cards. Each IO card provides serial and parallel connections, two fast-wide differential SCSI-2 channels, and space for two HIO controller cards. An HIO controller card offers a HIPPI card, a FDDI card, and a card supporting three SCSI-2 channels.

The number of IO cards specified for each Ingest Server is determined by allocating HIO slots to the FDDI and HIPPI interfaces, and counting the number of SCSI-2 interfaces required. The number of internal and external SCSI-2 devices supported by the system determines the number of SCSI-2 interfaces required. The first SCSI-2 channel is delegated to internal devices like CD-ROMs, floppy disk drives, and tape drives. Internal disks ranging in aggregate size from four GB to eight GB are allocated to the second SCSI-2 channel. External disk arrays are allocated to subsequent SCSI-2 channels and the number of channels is based on the required throughput of the external file systems (See 920-TDx-001 and 922-TDx-009 series of base-line documents).

The internal disks on an Ingest Server are used to provide swap space for the operating system and to provide file space for the operating system and applications (See 920-TDx-001 and 922-TDx-009 series of base-line documents).

The external disk arrays are SCSI-2 based RAID units from SGI. The implementation of these ingest buffer arrays are RAID Level 3 and RAID Level 1 for the database and logs (See 922-TDx-009 series of base-line documents). The SGI RAID units use one redundant disk for each four data disks in the RAID Level 3 configuration, and these are built into groups of five disks, with up to four groups (20 disks) per RAID enclosure.

Each enclosure contains two controllers that can access one or more groups of disks in the enclosure. Both controllers can access a group of disks within an enclosure, however, only one controller can access the group at a time. This dual attachment enables the striping of disk volumes across controllers for high throughput and for implementing fail-over of controllers.

A FDDI subnetwork is implemented at each site. Each server of ICLHW is dual-attached to the SDPS FDDI subnetwork (See 920-TDx-001, 921-TDx-002, 921-TDx-003, and 921-TDx-004 series of base-line documents).

A central HIPPI switch HIPPI network. is used to implement the ICLHW and Data Repositories with switched 800 Mbps interface ports connected directly to the ICLHW and Data Repository hosts (See 920-TDx-001, 920-TDx-002, 921-TDx-003, AND 921-TDx-004 series of base-line documents).

Recovery/fail-over for HWCIs is described in the 920-TDx-014 series of documents. There is a version for each DAAC, indicated by the letter appearing in place of the "x." The document provides the recovery procedure for each host

#### **4.2.2.2 Ingest Workstation Description**

The Ingest Workstation is provided to execute the Ingest GUI. This workstation enables the operator to remotely monitor the Ingest Servers and the Ingest processes, from media ingest to remote ingest.

The Ingest Workstation is a 64-bit SGI machine. For information on the processor used on the Ingest Workstation see the 920-TDx-001 series of base-line documents.

The Ingest Workstation is equipped with at least 128 MB of memory (See 920-TDx-001 series of base-line documents).

The Ingest Workstation is equipped with four EISA slots, with a transfer rate of at least 33 MB/second. A FDDI interface card and a graphics subsystem each use an EISA slot. Additionally, the Ingest Workstation is equipped with two fast SCSI-2 connections.

The Ingest Workstation internal disks, providing an aggregate space of eight GB, provides swap space and file system space for the operating system and applications (See 920-TDx-001 series of base-line documents). There are no external disk arrays.

A FDDI subnetwork is implemented at each site. The Ingest Workstation uses a single-attached FDDI interface to connect to the SDPS FDDI subnetwork.

Recovery/fail-over for HWCIs is described in the 920-TDx-014 series of documents. There is a version for each DAAC, indicated by the letter appearing in place of the "x." The document provides the recovery procedure for each host.

### 4.3 Client Subsystem Overview

The Client Subsystem (CLS) is a set of CSCIs and processes to access the services and data available in ECS and the other systems interoperable with the ECS. The CLS also includes the services needed to implement an ECS interface for an application such as automated or custom data access. An accessed service can be remote (i.e., via wide-area network to other sites) or local (i.e., to a database manager at the user's site).

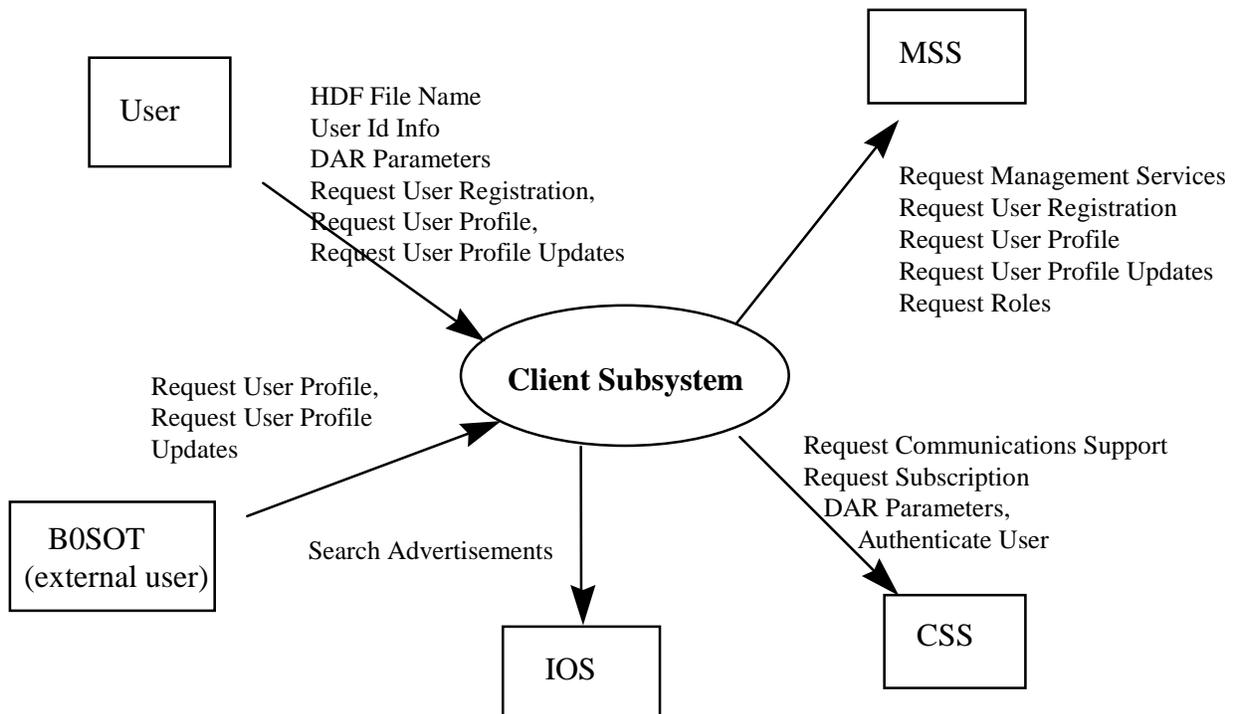
The CLS CSCIs and processes fall into the following general categories:

- An intuitive desktop manager to manage desktop objects (files) in the user's local file space, and represent applications and data in the domain of the client (applications in addition to the those available in CLS).
- Application programs accessible through user interfaces to implement the range of functionality available in the CLS.

In addition, the workstations operating within an ECS CLS contains infrastructure support software as part of the CSS and platform support software.

#### Client Subsystem Context

Figure 4.3-1 is the Client Subsystem context diagram. The diagram shows the events sent to the CLS and the events the CLS sends to other ECS subsystems.



**Figure 4.3-1. Client Subsystem Context Diagram**

The search and retrieval of data are performed by the B0SOT, the Version 0 Client (Motif version), ported to the ECS environment. The B0SOT is treated as an external entity since the design and design documentation are controlled under another contract. Table 4.3-1 provides descriptions of the interface events shown in the Client Subsystem context diagram.

**Table 4.3-1. Client Subsystem Interface Events (1 of 2)**

Event	Interface Event Description
Search Advertisements	The CLS requests a search of advertisements from the Advertising Service (within the Interoperability Subsystem) using the Advertising Service Application Program Interface (API) to obtain service signature information and the Universal References for the servers it contacts. In response, the CLS receives a list of advertisements.
Request User Registration	Users submit a request to be a registered user of the ECS. This enables special privileges not awarded to guests, such as the ability to order data that could cost money (such as ordering data on media that comes at a cost). The request is transferred to the MSS for processing. In response, the CLS receives a user registration response that the request has been received. The user through other means receives the actual account information.
Request User Profile	A User Profile Request is a search for a User Profile. There are two methods used, one is by an encrypted user name and password and the other is by user identification (ID). In response, the CLS receives the user profile, which contains information about a user that must be maintained. This includes mailing, billing, and shipping addresses, phone number, electronic mail address, etc. In response, the CLS receives a user's profile information for viewing or updating.
Request User Profile Updates	The user can update their User Profile through the B.0 Search and Order Tool (BOSOT). This includes their addresses (user, shipping, billing, and e-mail) and other important information. This updated profile information (profile2.odl) is forwarded through the CLS to the MSS. The BOSOT uses the profile2.odl file rather than a live interface with the CLS as its source of user information.
Request Roles	A request for roles of the operator. When an operator starts the DAAC Desktop CSCI, the Desktop CSCI requests the list of roles the operator can play from the MSS. The operator can select which role is being played at the time of logging in to the Desktop. The result of a Role request is a list of roles an operator plays. For example, the user might be a science coordinator (Sci_Coord) or a user services representative (User_Rep).
User ID Info	A user name and password for authenticating the user. The user name and password is turned into a User Authentication Request to the CSS (via a request for communications support).
DAR Parameters	Parameters required for submittal of Data Acquisition Requests (DARs) as specified in the ASTER GDS Interface Requirements Document (IRD)/Interface Control Document (ICD). As the result of a DAR submission, the user receives a DAR ID. This is a string of characters used to track a DAR. The user receives notification every time data resulting from this DAR is received by the system.
HDF File Name	This is a file name for a Hierarchical Data Format (HDF) file. The user opens the file to see the data in the file.

**Table 4.3-1. Client Subsystem Interface Events (2 of 2)**

Event	Interface Event Description
Request Subscription	A request for notification upon a specific event occurring in the system. An example is subscribing to the insert of a particular granule type. A valid subscription request results in the return of a subscription identifier. The subscription Identifier is not returned to the user.
Request Management Services	The MSS provides a basic management library of services to the subsystems, implemented as client or server applications, using the CSS Process Framework. The basic management library of services include: <ul style="list-style-type: none"> <li>• <b>Lifecycle commands</b> - The MSS forwards commands to managed hosts in the network to start and to stop applications. On startup, it passes a parameter identifying the mode (e.g., OPS, SHARED, test, training) for the application to run.</li> </ul>
Request Communications Support	The CSS provides a library of services available to each ECS subsystem. The services required to perform the specific subsystem assignments are requested by the subsystem from the CSS. These services include: Distributed Computing Environment (DCE) support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode information.
Authenticate User	A request to authenticate the user given a user name and password. This uses the DCE services to authenticate the user. In response, a user authentication response is sent back to inform the DESKT whether the user name and password are valid in the DAAC DESKT.

### Client Subsystem Structure

The CLS consists of two CSCIs:

- The Desktop (DESKT) CSCI provides a standard GUI framework for the user's interaction with the ECS software and data. There are two versions of the DESKT, one for science users and one for DAAC operations personnel. The Science User DESKT does not use DCE as its authentication mechanism while the DAAC User Desktop does use DCE for authentication and authorization. The User Profile Gateway is a server in the DESKT CSCI that provides User Profile information to the Science User Desktop. The DESKT is located on each user's workstation at each DAAC.
- The Workbench (WKBCH) CSCI includes the set of ECS applications and libraries that provide access to the ECS data and services. In Release 4, there are three tools: the User Registration Tool (URT), the EOSView, and the DAR Tool. The URT is an HTML based tool. The DAR Tool and the EOSView are X/Motif applications resident on a science user's workstation.

The CLS contains no HWCIs. The DMS hardware (Interface Servers) provides the processing and storage for the WKBCH software.

The Interface Servers are SUN Ultra 3000 Server class machines. Detailed specifications can be found in the site specific hardware design diagram, 920-TDx-001. Because of their common configuration, these servers can be used interchangeably. The Workbench software executed on these hosts enables user access to the ECS data and services. The Interface Servers also provide storage for user session data and the JAVA Earth Science Tool (JEST) map data. User session data is considered critical and thus stored in a Redundant Array of Inexpensive Disks (RAID) for high availability.

Detailed mappings can be found in the site-specific hardware/software mapping in baseline document number 920-TDx-002.

A SUN SPARC Storage Array Model 114, is dual ported between both hosts and provides storage for the user session data as described. A detailed configuration is specified per disk partition, and can be found in baseline document number 922-TDx-009.

Custom code and client applications are loaded on the internal disks of all hosts to prevent dependencies on specific hosts or peripherals. Real time or transaction oriented data (i.e., user session data) is stored in the RAID. The Interface Servers are both “hot” and share the resident RAID device. In the event of a host failure, the operational server assumes total ownership of the RAID and its processes. While in this state, the operational server is recognized to be running in degraded mode until recovery is completed.

#### **Use of COTS in the Client Subsystem**

- Netscape Navigator

The Netscape Navigator Web browser accesses the CLS inside a DAAC. The users can use the Web browsers they already have at their facilities.

- Netscape Enterprise Server

The Netscape Enterprise Server is used to serve the Web pages for the User Registration Tool.

- Environmental Systems Research Institute’s (ESRIs) ArcInfo

The ArcInfo package provides the spatial capability of the DAR Tool. The package provides a shared library and ArcInfo must be installed with the DAR Tool for the user to draw shapes on the map.

- XVT Software Inc.’s XVT

The XVT is used as a widget set and development tool for the EOSView application of the WKBCI CSCI. There is no operational unit of XVT used at run-time.

- Interactive Data Language (IDL)

IDL is used by EOSView to provide the visualization features for users.

- X/Motif  
X/Motif software was installed with the operating systems used for the DAR Tool and the DESKT respectively. These tools are currently available on Sun platforms only.
- Integrated Computer Solutions' (ICS) Builder Xcessory  
Builder Xcessory was used to design the initial layout of the X-Motif GUI screens for the X-Motif version of the DAR Tool. The X-Motif GUI screens were used to begin the layout of the current (JAVA) DAR Tool GUI screens.
- RogueWave's Tools.h++  
The Tools.h++ class libraries provide strings and collections to the DAR Tool and the User Registration Tool.

### 4.3.1 Desktop Software Description

#### 4.3.1.1 Desktop Functional Overview

The DESKT CSCI provides a desktop environment, consisting of a set of user and CLS software files (called desktop objects, each with a representative icon). A pop-up menu is provided for quick access to specific client applications, specific operations defined for desktop objects, including drag and drop, double-click to open, copy, delete, and rename. A desktop manager is provided to manage the set of desktop objects. Desktop objects include, but are not limited to, documents, data granule references, project folders, application programs, and configuration files. Managing the desktop objects includes viewing them in a hierarchical format and viewing their attributes, such as name, creation date, and time. The three types of desktop objects are: Container, Application, and Document.

The DESKT has two versions, the Science User DESKT and the DAAC User DESKT. The difference between the two versions is the Science User DESKT does not require the workstation it runs on to have a DCE Client. The Science User DESKT communicates via a User Profile Gateway over plain sockets to authenticate the user and retrieve user information. (The User Profile Gateway is a CSC in the DESKT CSCI.) The DAAC User DESKT uses DCE to authenticate the user. The DAAC user DESKT communicates directly with the MSS user registration server to retrieve user information.

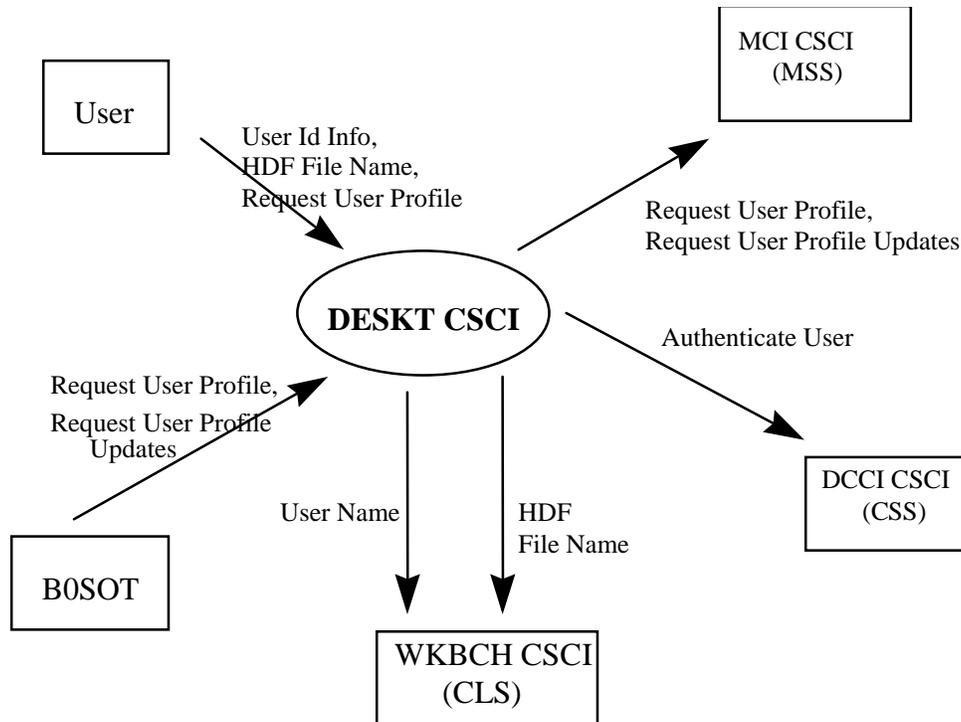
The DESKT is represented in either hierarchical or iconic format. The hierarchical format is similar to the Microsoft Windows implementation of the File Manager or the Macintosh Desktop implementation using View-By-Name. The iconic format is similar to the Microsoft Windows Program Manager implementation or the Macintosh Desktop using View-By-Icon mode. The DESKT contains desktop objects. Each desktop object has zero or more actions attached to it. The default action for an object is executed by double clicking on the icon that represents the object. Other actions can be performed from the pull-down menu. For example, an icon representing a directory would have a default action of being opened with the DESKT window (Open-In-Place). A secondary action for the default action is to open the directory in a new window (Open-New- Window). Selecting the icon and using the pull-down menu to open the

directory in a new window activates this secondary action. Using the DESKT can create these secondary actions.

Users can add their own tools to the desktop by modifying the appropriate DESKT configuration files. Configuration files must be modified to create an object type and specify the appropriate actions as well as Drag and Drop specifications. The files are similar to the X/Open Common Desktop Environment (CDE) specification.

#### 4.3.1.2 Desktop Context

Figure 4.3.1.2-1 is the DESKT CSCI context diagram. The diagram shows the events sent to the DESKT CSCI and events the DESKT CSCI sends to other CSCIs. Table 4.3.1.2-1 provides descriptions of the interface events shown in the DESKT CSCI context diagram.



**Figure 4.3.1.2-1. DESKT CSCI Context Diagram**

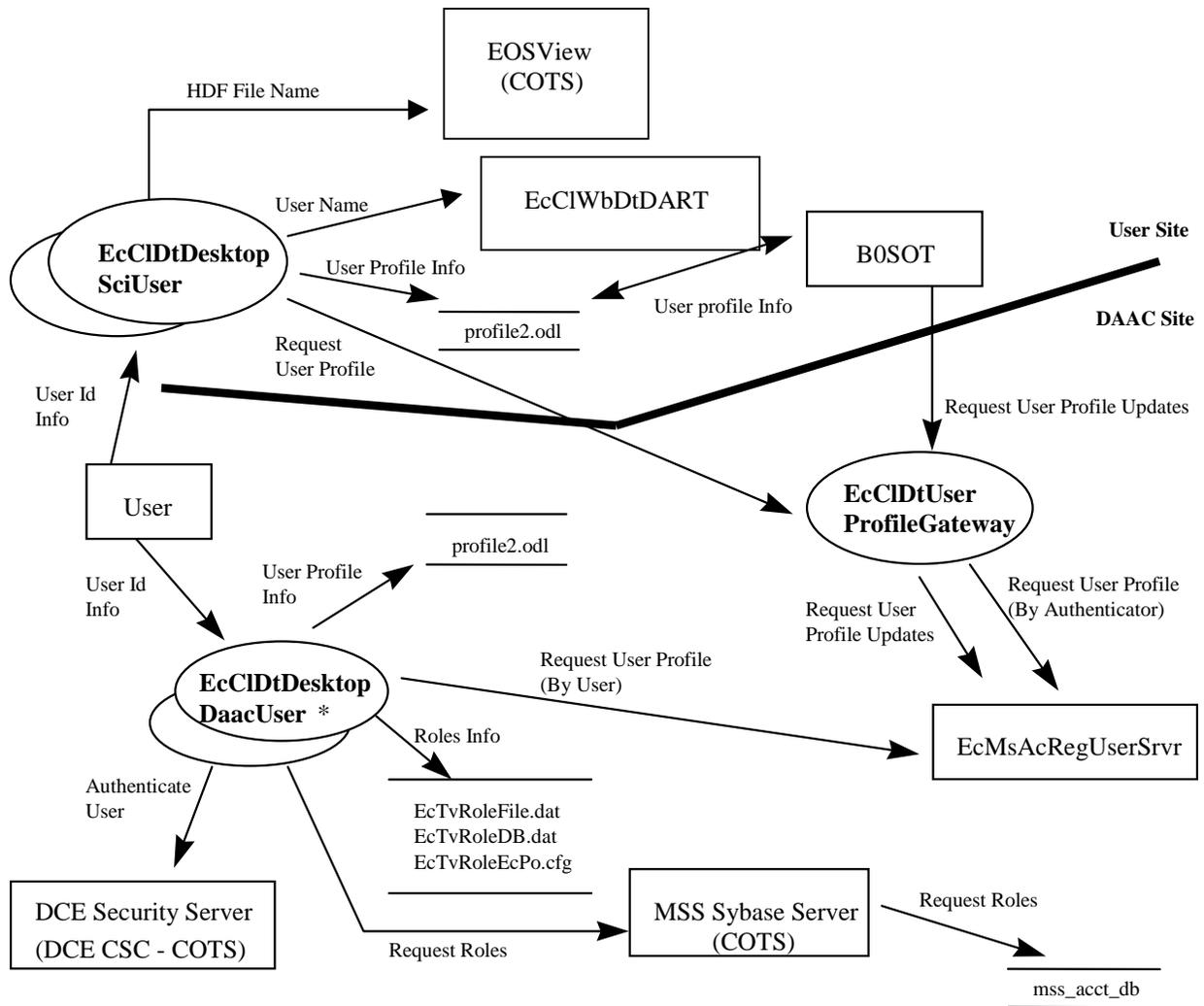
**Table 4.3.1.2-1. DESKT CSCI Interface Events**

Event	Interface Event Description
Authenticate User	A request to authenticate the user given a user name and password. This uses the DCE services to authenticate the user. In response, a user authentication response is sent back to inform the DESKT whether the user name and password are valid in the DAAC DESKT.
Request User Profile	This is a request to the MCI CSCI for retrieval of a User Profile. The user profile is the important information about a user that must be maintained. This includes, but is not limited to, mailing, billing, and shipping addresses, phone number, and electronic mail address. User Profiles can be retrieved either with the ECS userID or the ECS Authenticator. The Authenticator is an encrypted version of the userID and password. It is encrypted so it can be passed over a socket without threat of being stolen.
Request User Profile Updates	Users can update their User Profile information through the BOSOT. This includes their addresses (user, shipping, billing, and e-mail) and other information. The BOSOT uses this file rather than a live interface with the DESKT CSCI as its source of user information. This file gets forwarded to the DESKT CSCI and the DESKT CSCI forwards the file to the MCI CSCI.
User ID Info	This is a user name and password used for authenticating the user.
User Name	The user name is passed from the DESKT CSCI to the WKBCH CSCI.
HDF File Name	The name of an HDF file the user wants to see. It is passed from the DESKT CSCI to the WKBCH CSCI by dragging an icon to a Workbench application.

### 4.3.1.3 Desktop Architecture

Figure 4.3.1.3-1 is the DESKT CSCI architecture diagram. The diagram shows the events sent to the DESKT CSCI processes and the events the DESKT CSCI processes send to other processes.

The DESKT CSCI consists of three processes. Two of the processes are on the user's workstation where a user is either a science user or a DAAC operator. There are as many physical processes as there are users and operators in the system. The third process is the User Profile Gateway, a server that listens for calls on a socket from the Science User DESKT. This process resides inside the DAAC on the INTHW-1 server.



\* DAAC Desktop has same interfaces with WKBCH and B0SOT as the Science User Desktop.

**Figure 4.3.1.3-1. DESKT CSCI Architecture Diagram**

#### 4.3.1.4 Desktop Process Description

Table 4.3.1.4-1 provides descriptions of the processes shown in the DESKT CSCI architecture diagram.

**Table 4.3.1.4-1. DESKT CSCI Processes (1 of 2)**

Process	Type	COTS/ Developed	Functionality
EcCIDtDesktopSciUser	GUI	Developed	This is the end-user DESKT intended for science users. It is independent of DCE allowing users without DCE to install it on their local workstation. It contains the desktop manager and a set of default desktop objects, such as icons to the User Registration Tool, Advertising Service, etc. Once the process is installed, the user is allowed to add/delete icons from the DESKT.
EcCIDtDesktopDaacUser	GUI	Developed	This is the DESKT for the DAAC operations staff. It uses DCE to authenticate the user. The DCE login is expected to be the same as the user's UNIX login. It contains the desktop manager and a set of default desktop objects, such as icons that come with each Science User DESKT and icons applicable to the role of the operator. All the operator roles contain the tools required for the applicable role of the operator. Once the process is installed, the user is allowed to add/delete icons from the DESKT.
EcCIDtUserProfileGateway	Server	Developed	<p>The User Profile Gateway is a retrieval engine for the Science User DESKT. This enables the Science User DESKT to be independent of DCE. The server listens for calls on a socket from the DESKT CSCI.</p> <p>Interfaces:</p> <ul style="list-style-type: none"> <li>• User Profile Request: Request for a user profile given the user's authenticator</li> <li>• Profile Update Request: Request to update the user's profile with new information.</li> </ul> <p>Server Supports:</p> <ul style="list-style-type: none"> <li>• Multiple concurrent requests.</li> </ul>
EOSView	GUI	COTS	This is the end user tool that provides data visualization functions for the ECS data. It is used by general users of the system and personnel within the DAAC and SCF for Quality Assurance (QA) checks of products.

**Table 4.3.1.4-1. DESKT CSCI Processes (2 of 2)**

Process	Type	COTS/ Developed	Functionality
EcCIWbDtDART	GUI	developed	This is a X/Motif GUI that enables users to submit DARs to the ASTER GDS through the CSS DAR Communications Gateway. When the DAR is submitted, a DAR Identifier is returned to the user. A subscription is submitted on behalf of the user, asking for notification whenever a data granule with the specified DAR ID is inserted into the ECS archives. The DAR Tool also has the capability to modify DARs in accordance with the ASTER ICD.

#### 4.3.1.5 Desktop Process Interface Descriptions

Table 4.3.1.5-1 provides descriptions of the interface events shown in the DESKT CSCI architecture diagram.

**Table 4.3.1.5-1. DESKT CSCI Process Interface Events (1 of 2)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request User Profile	One per User Profile request	<i>Library:</i> MsAcCInt <i>Class:</i> MsAcUsrProfile Rwnet.h* RWSocketPortal	<i>Process:</i> EcCIDtDesktopSciUser	This is a request for the User Profile information based on an ECS "Authenticator". The user name and password entered is encrypted into a string. This string is passed to the User Profile Gateway. The MSS User Profile uses this string to find the user's profile. In response, the user profile is sent back to the DESKT process that requested the profile.
Request User Profile Updates	One per User Profile request	<i>Library:</i> MsAcCInt <i>Class:</i> MsAcUsrProfile Rwnet.h* RWSocketPortal	<i>Process:</i> EcCIDtUserProfileGateway	Users can update their profiles on the BOSOT. This interface to the EcCIDtUserProfileGateway requests an update to the user's profile.
Request User Profile (by Authenticator)	One per User Profile request to MSS	<i>Library:</i> MsAcCInt <i>Class:</i> MsAcUsrProfile Rwnet.h* RWSocketPortal	<i>Process:</i> EcCIDtUserProfileGateway	This is the request to the EcMsAcRegUserSvr using the ECS Authenticator.

**Table 4.3.1.5-1. DESKT CSCI Process Interface Events (2 of 2)**

Event	Event Frequency	Interface	Initiated By	Event Description
Request User Profile (by User)	One per User Profile request with user id	<i>Library:</i> MsAcClnt <i>Class:</i> MsAcUsrProfile Rwnet.h* RWSocketPortal	<i>Process:</i> EcCIDtDesktopDaacUser	This is the request to the EcMsAcRegUserSrvr using the user's ID.
User Profile Info	One per User Profile store to file	<i>Library:</i> C++ library <i>Class:</i> ofstream	<i>Processes:</i> EcCIDtDesktopSciUser EcCIDtDesktopDaacUser	The User Profile information is stored locally in a flat file after retrieval and return to the DESKT processes.
Request Roles	One per query for operator role data	<i>Library:</i> EcTvRoles <i>Class:</i> EcTvRoleManager	<i>Process:</i> EcCIDtDesktopDaacUser	The Request Roles is a query to the MSS database to retrieve the operator's role. The EcCIDtDesktopDaacUser calls a MSS API, which does the database retrieval. The roles information is returned to the EcCIDtDesktopDaacUser who requests the information to be stored in files.
User ID Info	One per user id	Text typed in by user.	User (a person)	The User ID Info is a user name and password supplied by the user.
HDF File Name	One per HDF file selected	Read from .ECS directory	<i>Processes:</i> EcCIDtDesktopSciUser EcCIDtDesktopDaacUser	The HDF File is a file name selected by the user from the objects on the DESKT.
User Name	One per user name passed to Workbench	Either environment variables or URL as applicable.	<i>Processes:</i> EcCIDtDesktopSciUser EcCIDtDesktopDaacUser	The user name entered on the desktop login screen is passed to tools in the Workbench CSCI.
Authenticate User	One per authentication request	<i>Library:</i> EcSeLoginSh <i>Function:</i> EcSeChkDcelogin	<i>Process:</i> EcCIDtDesktopDaacUser	The authentication request is a request to the DCE Security Server, of the DCCI CSCI, to check the login of the user entered on the DAAC Desktop.
Roles Info	One per role request	<i>Library:</i> EcTvRoles <i>Class:</i> EcTvRoleManager	<i>Process:</i> EcCIDtDesktopDaacUser	The Roles Information (the tools applicable for a role) is requested from the MSS accountability database and returned to the DESKT.

### 4.3.1.6 Desktop Data Stores

Table 4.3.1.6-1 provides descriptions of the DESKT CSCI data storage areas. To simplify the table, the list of data stores is limited to the areas shown.

**Table 4.3.1.6-1. DESKT CSCI Data Stores**

<b>Data Store</b>	<b>Type</b>	<b>Functionality</b>
profile2.odl	File	This file is where the DESKT writes the User Profile information. This is required since B0SOT does not have a "live" link to the User Profile Gateway to retrieve it. The DESKT writes it to the file and B0SOT uses the file upon startup.
mss_acct_db	Database	The DESKT CSCI uses a MSS API to retrieve role information from the MSS database. This connection must be made for the DESKT CSCI to operate.
EcTvRoleFile.dat	File	This file contains the cross-reference of the user name to the user's role. The role is used to go to the MSS API to retrieve the tools applicable to the role.
EcTvRoleDB.dat	File	This file contains the application ID and Application Name for role retrieval.
EcTvRoleEcPo.cfg	File	This file contains parameters used by the MSS API to connect to its database.

## 4.3.2 Workbench Software Description

### 4.3.2.1 Workbench Functional Overview

The Workbench (WKBCH) CSCI is a set of application programs, which implement the core functionality of the CLS science user interface. The search and retrieval of data are performed by the B0SOT, Version 0 Client (Motif version), ported to the ECS environment. The B0SOT is treated as an external entity since the design and design documentation are controlled under another contract. The WKBCH CSCI provides users the capability to register as users of the system, submit data acquisition requests for the ASTER instrument data, and to see data products in HDF format.

The User Registration Tool (URT) handles requests by users to become registered users of the ECS. URT requests specific mailing address, E-mail address, and phone number to pass to the MSS for processing. The user is contacted directly to receive the ECS login information. After the submission of the user registration request, the CLS is not involved in the rest of the transaction.

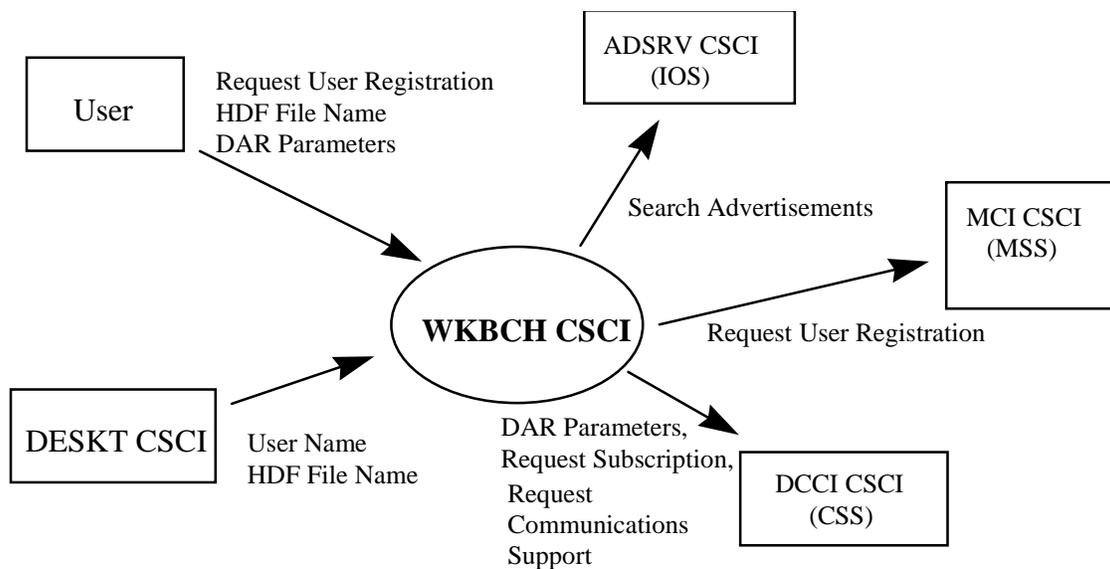
The DAR Tool handles user requests to acquire ASTER data from the satellite. The user specifies when and where the data is to be taken. The DAR is submitted to the ASTER GDS in Japan for review. The acceptance of the DAR by Japan is not immediate and the DAR Tool can not determine the DAR status immediately. The DAR Tool is a X/Motif GUI that currently only runs on a Sun workstation with DCE installed.

EOSView is an HDF-EOS viewer that enables users to visualize data they receive from the ECS. EOSView can take any HDF-EOS data file and perform visualization functions while it does not provide sophisticated data analysis functions found in COTS products such as IDL. EOSView is a GUI application used on UNIX platforms using X/Windows and Motif.

Each tool is integrated with the DESKT. HDF files can be dragged and dropped on the EOSView icon to initiate the tool with the specified HDF file.

#### 4.3.2.2 Workbench Context

Figure 4.3.2.2-1 is the WKBCH CSCI context diagram. The diagram shows the events sent by WKBCH to other ECS subsystems and events sent to WKBCH from the other ECS subsystems.



**Figure 4.3.2.2-1. WKBCH CSCI Context Diagram**

Table 4.3.2.2-1 provides descriptions of the interface events shown in the WKBCH CSCI Context Diagram.

**Table 4.3.2.2-1. WKBCH CSCI Interface Events (1 of 2)**

Event	Interface Event Description
Request User Registration	A user submits a request to be a registered user of the ECS. Registered users are given special privileges not awarded to guests, such as the ability to order data on a media that comes at a cost. There is an immediate response to the user that the User Registration Request was received by ECS. The user through other means receives the actual account information.

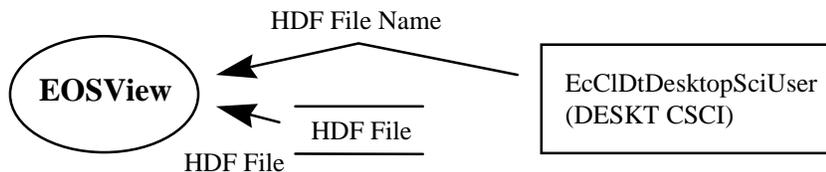
**Table 4.3.2.2-1. WKBCH CSCI Interface Events (2 of 2)**

Event	Interface Event Description
Search Advertisements	The CLS requests service advertisements via the Advertising Service (Interoperability Subsystem) API to obtain service signature information and the Universal References for the servers it contacts.
Request Subscription	A request for notification of a specific event occurring in the system is a subscription. An example would be subscribing to the insert of a particular granule type. A valid subscription request results in a subscription identifier. The subscription Identifier is not returned to the user.
DAR Parameters	Data Acquisition Requests (DAR) parameters are required for submittal of DARs as specified in the ASTER GDS IRD/ICD. As the result of a DAR, the user receives a DAR ID. This is a string of characters used to track a DAR. The user receives notification every time data resulting from this DAR is received by the ECS.
HDF File Name	This is the name of an HDF file to be opened in EOSView. A file name passed to EOSView enables EOSView to present the file contents to the user. This file can be obtained through any means (i.e., from a search, browse, acquire, or from a friend). The EOSView enables the user to manipulate an HDF file into multiple types of displays. Images, metadata, and actual data values can be viewed. Multiple images can be animated.
User Name	The user name that identifies the user to the ECS.
Request Communications Support	The DCCI CSCI provides a library of services available to each ECS CSCI. The services required to perform the specific CSCI assignments are requested by the CSCI from the DCCI CSCI. These services include: DCE support, file transfer services, Network & Distributed File Services, Bulk Data transfer services, file copying services, name/address services, password services, Server Request Framework (SRF), Universal Reference (UR), Error/Event logging, message passing, Fault Handling services, User Authentication services, and Mode Information.

**4.3.2.3 Workbench Architecture**

EOSView and the DAR Tool run on the user’s workstation. For the URT, a Web browser is used on the workstation to access Common Gateway Interface (CGI) scripts located inside the DAAC. Since these tools have no interfaces to each other and have distinct operations, their uses are described separately. Figure 4.3.2.3-1 is the EOSView architecture diagram. The diagram shows the events sent to the EOSView tool.

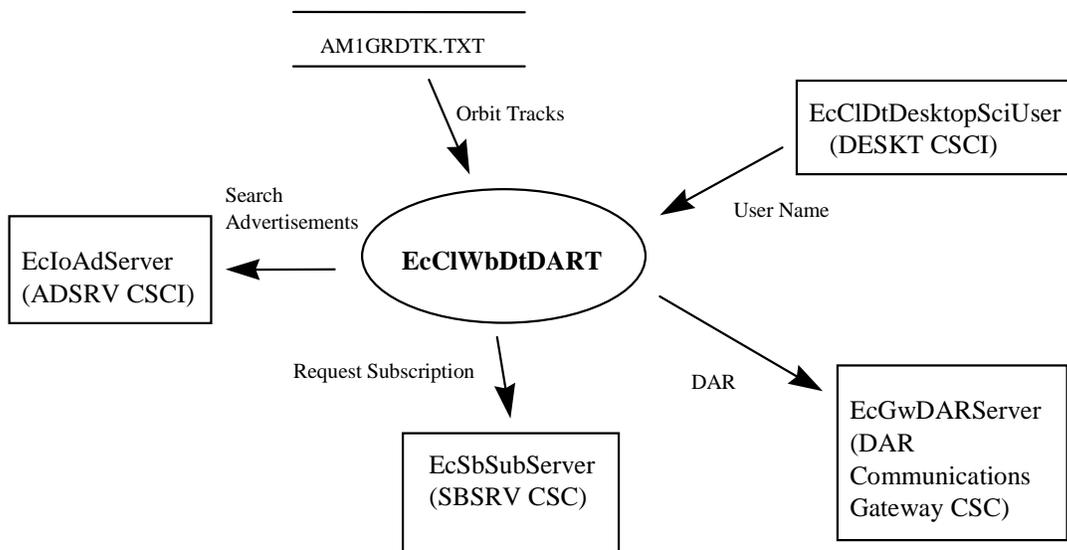
Since EOSView is a stand-alone application, it only interfaces with the DESKT CSCI and gets initialized with a file name or from the command line.



**Figure 4.3.2.3-1. EOSView Architecture Diagram**

Figure 4.3.2.3-2 is the DAR Tool architecture diagram. The diagram shows the events sent to the DAR Tool process and the events the DAR Tool process sends to other CSCIs, CSCs, or processes.

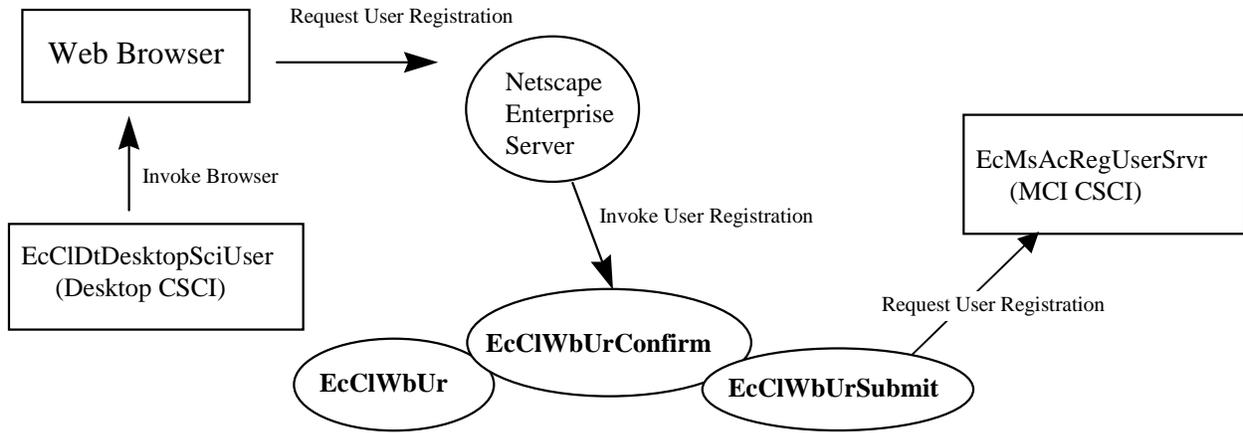
The DAR Tool reads a file of spacecraft orbital data. co-located with the DAR Tool on the user's workstation. The DAR Tool is initiated from the desktop and can be initiated from the command line. The DAR Tool submits DARs to the CSS DAR Communications Gateway CSC. DARs can be a DAR submission or a modification to an existing DAR. After the DAR is successfully submitted, the DAR Tool submits a subscription on behalf of the user to get notification when the data associated with the DAR is ingested into the ECS. The DAR Tool retrieves the service advertisement from the Advertising Service to determine the parameters to send to the Subscription Server.



**Figure 4.3.2.3-2. DAR Tool Architecture Diagram**

Figure 4.3.2.3-3 is the URT architecture diagram. The diagram shows the events sent to the URT processes and the events the URT processes send to other CSCIs, CSCs, or processes.

The URT is an HTML application and resides inside each DAAC. The HTML pages are served from the Netscape Enterprise Server located on the INTHW-1 server. The URT communicates with the MSS User Profile Server to submit the registration request.



**Figure 4.3.2.3-3. URT Architecture Diagram**

#### 4.3.2.4 Workbench Process Description

Table 4.3.2.4-1 provides descriptions of the processes shown in the EOSView, DAR Tool and the URT Architecture Diagrams.

**Table 4.3.2.4-1. WKBCH CSCI Processes**

Process	Type	COTS/ Developed	Functionality
EOSView	GUI	developed	This is the end user tool that provides data visualization functions for the ECS data. It is used by general users of the system and personnel within the DAAC and SCF for Quality Assurance (QA) checks of products.
EcCIWbUr, EcCIWbUrConfirm, EcCIWbUrSubmit	CGI	developed	These are the CGI scripts that display the URT user interface and provide the capability to submit a user registration request. There is a data entry screen, a confirmation screen, and finally a program to submit the request to the MSS.
EcCIWbDtDART	GUI	developed	This is a X/Motif GUI that enables users to submit DARs to the ASTER GDS through the CSS DAR Communications Gateway. When the DAR is submitted, a DAR Identifier is returned to the user. A subscription is submitted on behalf of the user, asking for notification whenever a data granule with the specified DAR ID is inserted into the ECS archives. The DAR Tool also has the capability to modify DARs in accordance with the ASTER ICD.

#### 4.3.2.5 Workbench Process Interface Descriptions

Table 4.3.2.5-1 provides descriptions of the process interface events shown in the EOSView, DAR Tool, and the URT Architecture Diagrams.

**Table 4.3.2.5-1. WKBCH CSCI Process Interface Events (1 of 2)**

Event	Event Frequency	Interface	Initiated By	Event Description
HDF File Name	One per user identified file	Read from .ecs directory	<i>Processes:</i> EcCIDtDesktopSciUser EcCIDtDesktopDaacUser	The user either drags an icon onto EOSView or types a file name on the EOSView user interface to tell the GUI which file to open.
HDF File	One file per request	File name provided by the user as read from the script file that captured it while active.	<i>Process:</i> EOSView	The file retrieved from the HDF File data store based on the file name provided by the user
User Name	One per ECS user	Either environment variables or URL depending on the application.	<i>Processes:</i> EcCIDtDesktopSciUser EcCIDtDesktopDaacUser	This is a request sent to the DAR Communications Gateway to be sent to the ASTER GDS.
DAR	One per request to DAR	<i>Class:</i> EcGwDARSubmit DarRequest_C	<i>Process:</i> EcCIWbDtDART	A request for notification upon a specific event occurring in the system. An example would be subscribing to the insert of a particular granule type. A valid subscription request results in the return of a subscription identifier. The subscription Identifier is not returned to the user. Note: This is the Request for Communications Support needed by the DAR Tool).
Request Subscription	One per notification request	<i>Library:</i> EcSbCI <i>Class:</i> EcCISubscription	<i>Process:</i> EcCIWbDtDART	The DAR Tool is used to search the Advertising Service (Interoperability Subsystem) to find the signature service advertisement for the subscription event related to the ASTER data. This provides the DAR Tool with the parameters needed to be filled in before sending the subscription request to the EcSbSubServer.
Search Advertisements	One per advertising search	<i>Libraries:</i> IoAdCore, IoAdSubs <i>Class:</i> IoAdApproved AdvSearchCommand	<i>Process:</i> EcCIWbDtDART	The DAR Tool is used to search the EcIoAdServer using the DAR Tool to find the signature service advertisement for the subscription event related to the ASTER data. This provides the DAR Tool with the parameters needed to be filled in before sending the subscription request to the EcSbSubServer.

**Table 4.3.2.5-1. WKBCH CSCI Process Interface Events (2 of 2)**

Event	Event Frequency	Interface	Initiated By	Event Description
Orbit Tracks	One per orbit track displayed	open/read standard Unix commands	<i>Process:</i> EcCIWbDtDART	The user to submit a Data Acquisition Request (DAR) displays the orbit tracks of the AM-1 spacecraft.
Request User Registration	One per user request	forklexec standard Unix calls	<i>Process:</i> EcCIDtDesktopSciUser	This is the request by the user to become registered as an ECS user.
Invoke User Registration	One per User Registration Request	forklexec standard Unix calls	<i>Process:</i> Netscape Enterprise Server	The Netscape Enterprise Server initiates the URT CGI scripts, passing in the data from the HTML forms the user completed in the User Registration Request.
Request User Registration	One per registration request to MSS	<i>Libraries:</i> MsAcCInt, MsAcComm <i>Classes:</i> EcAcUsrRequest Mgr MsAcUsrRequest	<i>Process:</i> EcCIWbUrSubmit	The URT CGI scripts take the user input and submit the registration request to the EcMsAcRegUserSrvr

#### 4.3.2.6 Workbench Data Stores

Table 4.3.2.6-1 provides descriptions of the WKBCH CSCI data storage areas shown on the EOSView, DAR Tool, and URT Architecture Diagrams. To simplify the table, the list of data stores is limited to the areas shown.

**Table 4.3.2.6-1. WKBCH CSCI Data Stores**

Data Store	Type	Functionality
AM1GRDTK.TXT	File	This is a listing of the orbit tracks for the AM-1 spacecraft used by the DAR Tool to display orbit tracks to the user during creation of a DAR. The data in this file are downloaded to the user's workstation during installation.
HDF File	File	A listing of the HDF files accessible by EOSView.